Efficient registration of stereo images by matching graph descriptions of edge segments

Nicholas Ayache, Bernard Faverjon

INRIA
CENTRE DE ROCQUENCOURT

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
BP 105
78153 Le Chesnay Cedex
France
Tél. (1) 39 63 55 11

Rapports de Recherche

N° 559

EFFICIENT REGISTRATION OF STEREO IMAGES BY MATCHING GRAPH DESCRIPTIONS OF EDGE SEGMENTS

Nicholas AYACHE
Bernard FAVERJON

Août 1986

EFFICIENT REGISTRATION OF STEREO IMAGES BY MATCHING GRAPH DESCRIPTIONS OF EDGE SEGMENTS

Nicholas AYACHE & Bernard FAVERJON

INRIA - Domaine de Voluceau
B.P. 105 - 78153 LE CHESNAY Cedex
FRANCE

Abstract

We present a new approach to the stereo-matching problem which uses a graph-based description of line segments in the images and a search strategy based on prediction and propagation of hypotheses. This approach was tested on a variety of man made environments, and it appears to be fast and robust enough for mobile robot navigation and 3D part positioning applications.

Key words : Stereo Vision, Edge Matching, Graph-Search, Prediction and Propagation, Mobile Robots.

MISE EN CORRESPONDANCE RAPIDE D'IMAGES STEREOSCOPIQUES PAR APPARIEMENT DE GRAPHES DE SEGMENTS DE CONTOURS

Résumé

Nous présentons une solution originale au problème de la mise en correspondance d'images stéréoscopiques qui utilise une description des images par un graphe de segments de contours et une technique d'appariement par prédiction et propagation d'hypothèses. Cette méthode a été testée sur différentes scènes d'intérieur et elle s'est avérée être suffisamment rapide et robuste pour des applications telles que la navigation d'un robot mobile ou le positionnement d'objets tridimensionnels.

Mots clés : Vision Stéréoscopique, Mise en Correspondance de Contours, Recherche dans un Graphe, Prédiction et Propagation, Robots Mobiles.

developped by [BERTHOD-85] which combines a corner detector and a linear least-squares estimate to locate linear segments with higher accuracy. Each segment is stored with the following list of features :

Segment [INDEX,x,y,LENGTH,ANG].

Where :

. INDEX is an integer which identifies the segment,
. x and y are the coordinates of the segment midpoint,
. LENGTH is the segment length,
. ANG is the orientation of the segment computed within the interval [-pi,pi],

These features are local and exclusively geometric measures except for the orientation feature which also contains implicitly the sign of the intensity difference (contrast) computed across the segment. It is possible to include additional local properties such as a measure of the average intensity or gradient, or a texture measure computed on each side of the segment.

II.3 - Neighborhood relations and buckets

In addition to the local features computed for each segment, we define some neighborhood relations between segments. To do so, we first define buckets of segments on the image. These buckets will be used not only to determine the neighborhood of segments at low cost, but also to reduce drastically the computing time devoted to the prediction and propagation of hypotheses (on the use of bucketing techniques see [KNUTH-75] for instance).

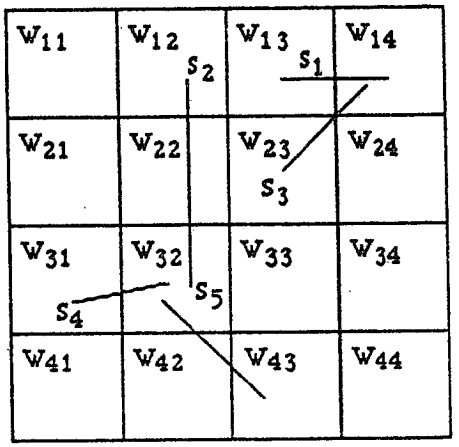
The computation of the buckets is performed by an algorithm whose complexity is only linear with respect to the number of segments in the image. The overall image is partitioned into m^2 squared windows W_i. To each window W_i is attached a bucket b_i which is a list of segments $\{S_j\}$ intersecting Wi. For each segment S_k of the image, the program computes the list of windows $\{W_i\}$ intersected by S_k. First this list is stored and attached to S_k and second, the segment S_k is appended to each bucket list b_i attached to the windows W_i (these lists are initially empty). When all segments S_k have

been considered, the process ends up with :

1) a list of intersected windows $\{W_i\}$ attached to each segment,

2) a list of intersecting segments $\{S_j\}$ attached to each window.

The neighborhood of a segment S_k is defined as the list of segments which intersect at least one common window with S_k. In practice, this list of neighbors is now simply obtained as the union of the buckets S_j attached to the windows W_i attached to S_k.This list of neighboring segments is actually computed for each segment and added to its description record. Figure 1 shows an example of computation of buckets and line segments neighborhoods.



list of intersected windows :
S_1 : $\{W_{13},W_{14}\}$; S_2 : $\{W_{12},W_{22},W_{32}\}$; S_3 : $\{W_{13},W_{14},W_{23},W_{24}\}$;
S_4 : $\{W_{31},W_{32}\}$; S_5 : $\{W_{32},W_{42},W_{43}\}$.

non empty buckets :
W_{12} : $\{S_2\}$; W_{13} : $\{S_1,S_3\}$; W_{14} : $\{S_1,S_2\}$;
W_{22} : $\{S_2\}$; W_{23} : $\{S_3\}$; W_{24} : $\{S_3\}$;
W_{31} : $\{S_4\}$; W_{32} : $\{S_2,S_4\}$;
W_{42} : $\{S_5\}$; W_{43} : $\{S_5\}$;

neighbors :
S_1 : $\{S_3\}$; S_2 : $\{S_4,S_5\}$; S_3 : $\{S_1\}$;
S_4 : $\{S_2,S_5\}$; S_5 : $\{S_2,S_4\}$;

Figure 1 : Computation of buckets and neighborhoods

Buckets will also be used to reduce the computing time devoted to the generation and propagation of hypotheses. As we shall see later, these processes require to match a segment within a given disparity window. By first computing the buckets which do intersect this window, one can substantially reduce the set of potential matching candidates to the union of these buckets. Concerning the window size, it has to be adapted to the disparity window ; to generate hypotheses, we use the relatively large (50×50 pixels^2) squared window based buckets used to compute the line segments neighborhoods. To propagate hypotheses, the disparity window is much narrower, and we have to compute another set of buckets based on vertical rectangular narrow windows. In these last buckets, the segments are sorted accordingly to their orientation, which is done at the expense of a unique global sorting of the whole set of segments. This sorting of buckets is also used to again speed-up the propagation process, as later explained.

III - STEREO MATCHING CONSTRAINTS

In this section, we express the constraints of the stereo matching problem. When the primitives to be matched are single points, the initial constraints are that homologous points must lie on corresponding epipolar lines and satisfy some local similarity constraints (intensity, gradient). To help solve the correspondence problem, global constraints are usually used : the uniqueness and the continuity constraints [MARR-76].

The uniqueness constraint states that a point in one image has at most one correct match in the other image. The continuity constraint expresses the fact that a correct match is generally surrounded by matches having similar disparities (disparity is defined as the difference in position between a pair of matched points).

We now extend these constraints to the case of line segment primitives.

III.1 - Local Constraints

We define potential homologous segments as being a pair (L,R) of segments of the left and right images verifying :

1: the epipolar constraint;

2: geometric similarity constraints;

III.1.1 - The Epipolar Constraint.

Given a point I_1 in one image, its homologous point I_2 is constrained to lie within a line in the other image called the epipolar line of I_1 (see Figure 2).

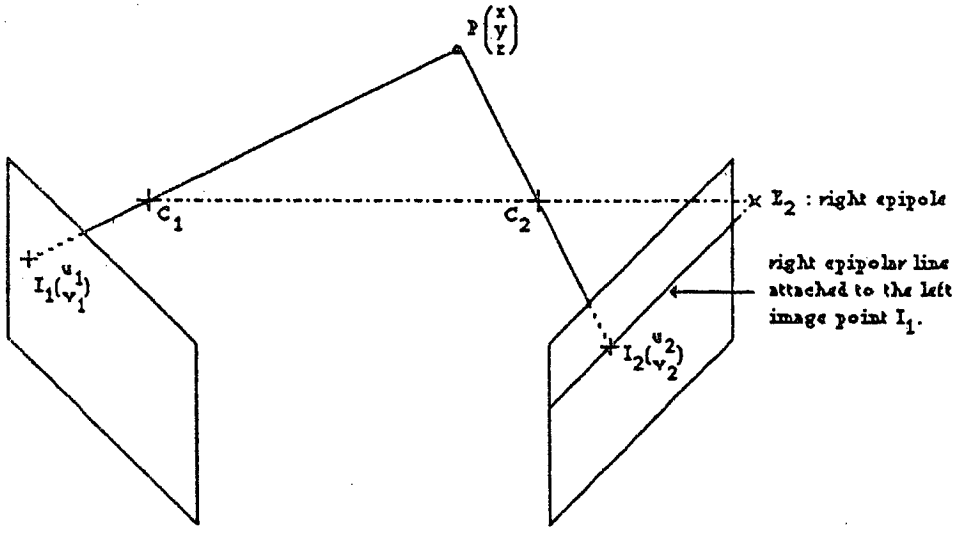


Figure 2 : Geometry of the epipolar lines in the right image : C_1 and C_2 are the optical centers of the cameras

Many authors make the assumption that the optical axes of the cameras are parallel in a plane orthogonal to the image planes, which constrains the search for homologous points along parallel image scan-lines. This constraint can be hard to achieve in practice. Also it can be desirable to make the optical axes to converge to increase the intersection of the fields of view of the cameras and/or improve the accuracy of the stereo reconstruction. Moreover, when the image planes are vertical, because roughly half of of the edges in indoor scenes are close to horizontal lines, it is highly desirable not to have epipolar lines parallel to the image scan-lines. Therefore we prefer to take into account the actual geometry of the stereoscopic system. It can be done by a very simple and efficient off-line calibration procedure described in the annex A. [FAUGERAS-86].

To extend the epipolar constraint from points to segments, we require that homologous segments contain at least a pair of homologous points. To simplify, we only require that the midpoint of L have an homologous point on R or that the midpoint

of R have an homologous point on L.

This definition is not symmetric, and differentiates left to right (L,R) and right to left (R,L) matches. The advantages of such a definition are twofolds :

- First, a given segment S has at most one correct match if we assume that the uniqueness constraint is satisfied for points (as there is only one homologous point corresponding to the midpoint of S).

- Second, homologous chains of edges which are approximated by a different number of segments in the two images can nevertheless be matched globaly by the union of left to right and right to left matches as it is shown in Figure 3 : here, for instance, we could find the left to right matches (L_1,R), (L_2,R) and (L_3,R) and the right to left match (R,L_2).

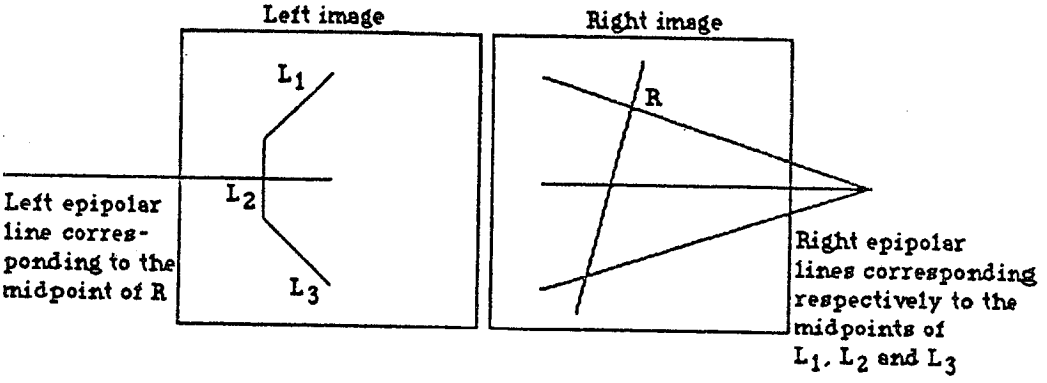


Figure 3 : Left segments L_1, L_2 and L_3 are matched to R while right segment R is matched to L_2

Disparity between potential homologous segments

Although the epipolar constraint is not symmetric, we define a symmetric measure of disparity DISP attached to a couple (L,R) of homologous segments belonging respectively to the left and right images. First, the program computes the common intersection of the segments with epipolar planes. This is done by computing the two epipolar lines in the right image corresponding to the endpoints of the left segment L. These lines intersect the line supporting the right segment R, yielding a segment R'. The intersection R" of R and R' is the set of points of R having potential homologous points on L. The corresponding set L" in L can also be computed. The

midpoints of L" and R" are homologous points, and the disparity DISP is defined by (cf. Figure 4) :

$$DISP = E_2I_R - E_1I_L$$

where E_2I_R (resp. E_1I_L) is the distance between I_R and the right epipole center, (resp I_L and the left epipole center)

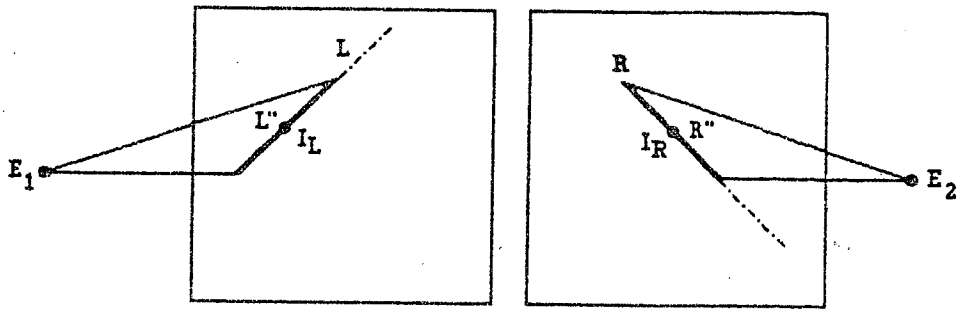


Figure 4 : Definition of disparity between L and R : $DISP = E_2I_R - E_1I_L$

III.1.2 - Geometric Similarity Constraints

The similarity measure consists in computing an error measure between the features LENGTH and ANG attached to L and R, and to reject the potential match (L,R) whenever one of these error measures is above a threshold.

These thresholds are set relatively tight during the prediction step and are loosened during the propagation step. They are computed in the following manner :

- Length is a relatively unstable feature. This is mainly due to the poor robustness of the polygonal approximation scheme. Typically a length ratio between homologous segments of 1.5 in the prediction step or 3 in the propagation step is tolerated.

- Angle is a robust feature with respect to the polygonal approximation scheme. Therefore the angular disparity between two homologous segments primarily comes from the viewpoint difference. A statistical study performed by [ARNOLD-80] showed that with narrow angle stereo -i.e. when the viewpoints are relatively close- the statistical distribution of the angular disparity between homologous segments (images of randomly oriented 3-D segments) is sharply centered close to zero. Therefore we set a tight threshold for the angular disparity (Typically 15 degrees during the prediction and

propagation steps).

III.2 - Global Constraints

III.2.1 - The Continuity Constraint

Up to now, we have only applied local geometric constraints to define potential homologous segments. We introduce now a global constraint known as the continuity constraint (cf. [MARR-76]): the observed scene is made of physical objects whose surfaces vary smoothly in general. Therefore, disparity between homologous points in the images should also vary smoothly in general, except at a few depth discontinuities. Hence, if a match (L,R,DISP) is correct, it is very likely that most of the neighbors L' of L (resp. R' of R) can find a match R' (resp. L') with a disparity close to DISP, which should not be the case when the initial match is incorrect. In other words, applying the continuity constraint to a given match will yield a large number of likely correct matches within the neighborhoods of L and R if the initial match is correct, and a small number of likely incorrect matches otherwise. If this process is repeatedly applied to all the new matches obtained, it should end up with a much larger final number of matches when the initial hypothesis is correct than when it is incorrect. This fact can be expressed in terms of large or small connected components of the disparity graph defined below.

Definition of the disparity graph

The disparity graph is defined as follows :

- nodes are couples (L,R) of potential matches between left and right images ;

- edges connect pairs of nodes (L,R), (L',R') such that L' and R' are respective neighbors of L and R in the image descriptions, and such that the disparity gradient between (L,R) and (L',R') (computed as the difference between the disparities attached to each match) is lower than a locally computed limit, called the *disparity gradient limit.*

The disparity gradient limit is computed as a *function of the position of the reconstructed corresponding 3D point*, and it is adapted to correspond to a *constant tolerated variation of depth* ε in space between 2 neighboring reconstructed segments.

The computation details are given in the annex B.

A connected component of the disparity graph corresponds to a subset of 3-D points which belong to what we call a *smooth surface patch* .

As an illustration of this definition, let us consider the example shown in Figure 5. There is a triangle lying below and in front of 2 rectangles. Figure 6 shows the neighboring graph produced in the 2 images. Figure 7 shows the disparity graph obtained for a given value of ε. There are 4 connected components.

The first one corresponds to the matching of the two rectangles against their respective homologues. The second one corresponds to the matching of the triangle. There is no connection between (3,3') and (9,9') or between (7,7') and (10,10') because the discontinuity in depth between the triangle and the rectangles is such that the corresponding 3-D points are at a distance greater than ε.

The third and fourth components of the disparity graph correspond to "phantom" images of the rectangles obtained by matching each of them with its *opposite* homologue.

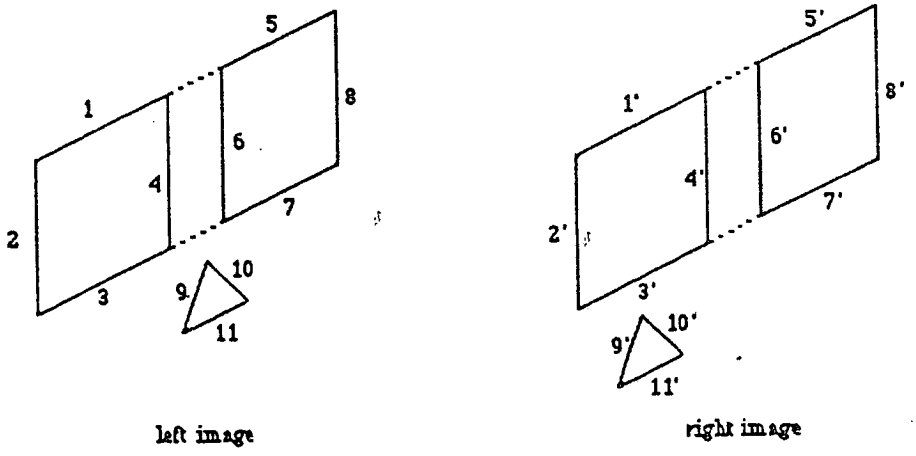


Figure 5 : the left and right images of a triangle lying in front of two rectangles

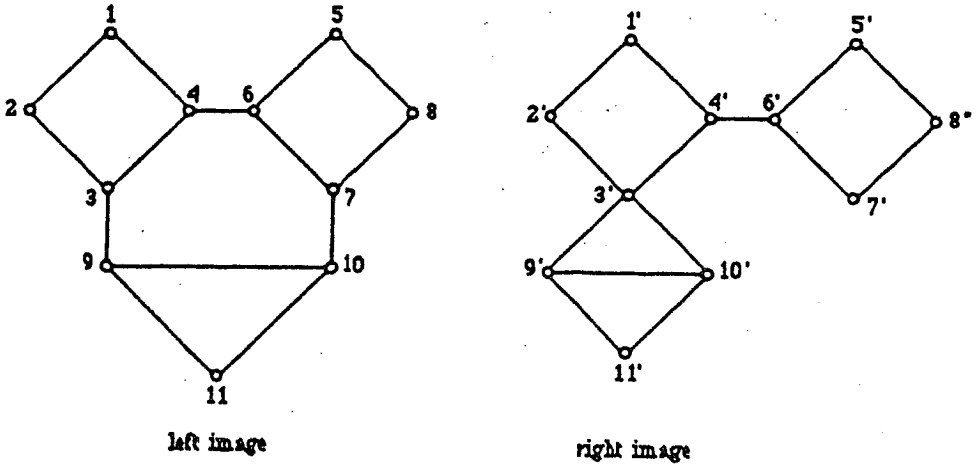


Figure 6 : neighborhood graphs

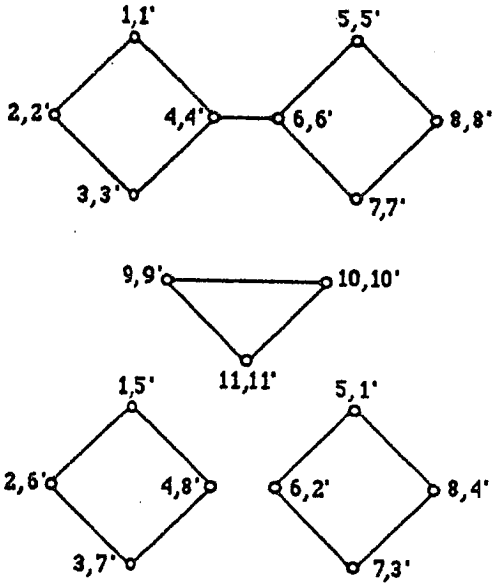


Figure 7 : disparity graph

III.2.2 - The Uniqueness Constraint

In order to enforce the uniqueness constraint, we must keep only one match per left or right segment. In case of conflict, the correct matches will usually correspond to the larger connected component. This gives us a criterion to choose between conflicting matches.

In the following sections we describe the practical algorithm used to achieve the stereo matching of line segments.

IV - PREDICTION AND PROPAGATION ALGORITHM

In order to compute the connected components of the disparity graph, we use a prediction and propagation algorithm. The idea is to generate some potential pairs of homologous segments based on a strong local similarity of contours and then to explore the attached connected components by a recursive propagation.

IV.1 - Prediction of hypotheses

Initially, the potential disparity range within the image is relatively large, and depends on the geometry of the imaging system and of the observed scene. (Typically the disparity range is about one third of the image width). When this interval has been estimated, the role of the prediction program is to establish a list of tentative matches (L,R) with associated disparities DISP lying within this interval and verifying the local constraints defined in the previous section.

In order to reduce the number of false matches, only segments which are neither too small (poor estimation of the orientation) neither too long (likely to be broken in the other image) and with orientations not too close to the orientation of the epipolar lines are considered.

The hypotheses are generated with some degree of uniformity over the whole image by randomly selecting the left segment L, and by stopping the prediction process when the number of matches is large enough. Also, the search for a right candidate R is substantially accelerated by the use of the squared buckets previously defined : given a left segment L, the program computes in the right image, the epipolar line corresponding to the midpoint of L, and on this line, the segment S_{disp} associated to the interval of potential disparities. The matching candidates R are then selected among the union of the right buckets intersected by the segment S_{disp}.

IV.2 - Propagation algorithm

At the end of the prediction stage, there is a certain number of potential matches. Each match corresponds to a single node in the disparity graph. The idea is to use the disparity graph to propagate these matches within their neighborhood to recover subsets of 3-D segments lying on a same smooth surface patch (that is matches which belong to

a same connected subgraph of the disparity graph).

A conflict is defined as a pair of matches (L,R) , (L,R') such that R≠R' or (L,R), (L',R) such that L≠L'. Conflicts occuring during the propagation stage are immediatly solved to avoid multiple matches within a same connected subgraph. Conflicts occuring between two distinct connected subgraphs are solved after the propagation of each connected component by analying they respective size (Power Of Prediction).

IV.2.1 - Exploration of connected components

The propagation algorithm starts from a prediction, that is a given node of the disparity graph. All the nodes connected to this node in the disparity graph are then recursively considered. In practice the disparity graph is constructed during this exploration by alternatively and recursively looking for a match for the neighbors of the left or right segments in the left or right images. The propagation process is achieved by the two following very simple recursive procedures, called PROPAGATE_LEFT and PROPAGATE_RIGHT :

```
Recursive procedure PROPAGATE_LEFT (segment_left, predicted_disparity)
        . if ALREADY_VISITED (segment_left) then return ;
        . (segment_right, actual_disparity) : =
                    MATCH_LEFT (segment_left, predicted_disparity) ;
        . if (segment_right = nil) then return ;
        . For each neighbor neighbor_right of segment_right do
                PROPAGATE_RIGHT (neighbor_right, actual_disparity) ;
          end for ;
end procedure PROPAGATE_LEFT ;

Recursive procedure PROPAGATE_RIGHT (segment_right, predicted_disparity)
        . if ALREADY_VISITED (segment_right) then return ;
        . (segment_left, actual_disparity) : =
                    MATCH_RIGHT (segment_right, predicted_disparity) ;
        . if (segment_right = nil) then return ;
        . For each neighbor neighbor_left of segment_left do
                PROPAGATE_LEFT (neighbor_left, actual_disparity) ;
          end for ;
end procedure PROPAGATE_RIGHT ;
```

To propagate a right to left hypothesis (R,L,DISP) the program calls the sequence PROPAGATE_LEFT (L,DISP) ; this procedure first checks whether this segment has been visited already. If yes, it stops, in order to prevent an infinite loop.

Otherwise the procedure MATCH_LEFT is called. MATCH_LEFT is a procedure very similar to the one used to generate hypotheses. It takes as input a left segment L and a predicted disparity PREDICTED_DISPARITY, and returns a match (L,R') with actual disparity ACTUAL_DISPARITY. To do so, this program selects among the right segments, those whose disparity with L is within the interval PREDICTED_DISPARITY+ $\Delta 1$, PREDICTED_DISPARITY+$\Delta 2$ where $\Delta 1$ and $\Delta 2$ are computed as a function of the position in space of the reconstructed 3D segment (see the annex B).

Here again, the use of precomputed narrow vertical buckets drastically reduces the number of candidates as potential matches in the right image. Also, the initial sorting of the buckets with respect to the segments orientations allows for a fast binary search within each bucket of the candidates whose orientations are compatible with the orientation of the left segment.

If several candidates remain, only the one with the disparity closest to PREDICTED_DISPARITY is retained. By doing this, we privilegiate segments belonging to the surface whose distance from the observer has the smallest variation (fronto-parallel surfaces). Alternative criterions could be used at this point, for instance choosing the match yielding the 3D segment closest in the scene to the previous one. Anyhow, the current criterion, which yields good results in our experiments, has the merit of being simple and computationally cheap.

If no match is found, the program MATCH_LEFT returns NIL. In this case the procedure PROPAGATE_LEFT stops, which means that it cannot propagate matches any further with the predicted disparity PREDICTED_DISPARITY and the tolerances $\Delta 1$ and $\Delta 2$ applied to left segment L.

When MATCH_LEFT returns a match R' and an actual disparity ACTUAL_DISPARITY, the procedure PROPAGATE_LEFT calls the symmetric procedure PROPAGATE_RIGHT for each neighbor of R' and with an updated predicted disparity ACTUAL_DISPARITY.

The result of this propagation process is, for each hypothesis, a set of matches (L,R,DISP) corresponding to what we called a smooth surface patch. In the example of Figure 5, the hypothesis (1,5') wold propagate within the third connected component of

the disparity graph of Figure 7 yielding the matches (2,6'), (3,7'), (4,8'), while the hypothesis (1,1') would propagate within the first component yielding the matches (2,2'), (3,3'), (4,4'), (5,5'), (6,6'), (7,7'), (8,8').

IV.2.2 - Pruning of hypotheses

Because the technique used to select a match between conflicting matches during the propagation process is a *best first choice* technique, the result of the propagation process might depend on the starting node chosen within each connected component.

Nevertheless, in practical experiments, it appeared that this dependency was minor, due to the computation of the disparity gradient limit as a function of the distance. As a consequence, in order to substantially reduce the computing time of the algorithm, it is desirable to propagate only one hypothesis by connected component of the disparity graph. This is achieved by removing after the propagation of an hypothesis, all the forthcoming hypotheses corresponding to a match already propagated.

In the example of Figure 5, if the hypotheses (2,2'), (4,4'), (6,6'), (8,8') are made by the prediction algorithm, then, after the propagation of anyone of these hypotheses, the others will belong to the propagated matches and therefore be discarded.

IV.2.3 - Conflicts between connected components

When a new hypothesis is propagated, the same algorithm applies, without considering the matches obtained by previous hypotheses. When the propagation is over, conflicts between distinct connected components are solved by discarding the conflicting matches attached to the concerned hypothesis having the smaller Power Of Prediction (number of matches). In the example of Figure 5, if a given hypothesis yields the matches {(1,1'), (2,2'), (3,3'), (4,4'), (5,5'), (6,6'), (7,7'), (8,8')}, and if a new hypothesis yields the matches {(5,1'), (6,2'), (7,3'), (8,4')}, then all the conflicting matches of the second hypothesis (in this case the four of them) will be discarded because they belong to a smaller connected component of the disparity graph (of size 4 instead of 8). As one can see, this is an efficient procedure to distinguish between real objects and "phantoms" in case of repetitive structures.

Due to the potential deletions occuring after each new propagation, the size of an hypothesis can change after the propagation of another hypothesis; nevertheless, the algorithm uses only the initial size of each hypothesis to compare conflicting matches. By doing this, the final result does not depend on the order in which the hypotheses are compared.

V - FINAL VALIDATION

When all the hypotheses have been propagated, some of the connected components have been totaly or partially erased by better ones. However, some false matches not conflicting with others (in the sense of the uniqueness constraint) may remain. In our experiments, such matches represents 5 to 10 per cent of the final matches. They usually correspond to isolated 3D segments. Thus they can be easily removed by keeping only matches which are connected to a minimum number of nodes in the disparity graph. This minimum number is set depending on the nature of the observed scenes. In our experiments, we rejected connected components of size smaller than four. In this case, the remaining false matches represents less than 2 per cent of the matched segments.

VI - EXPERIMENTAL RESULTS

The stereo-matching program has been tested on several various scenes including outdoors and indoors scenes as well as scenes with industrial parts. We present here the results of the matching process applied to 4 typical scenes which are called 1) room, 2) office, 3) corridor, 4) industrial part, and which are shown in Figures 8, 9, 10, 11 respectively.

VI.1 - Preprocessing

Pairs of 512×512 images are taken by a pair of standard cameras. Contours on each image are then extracted and approximated by linear segments. For the first three scenes, the edge extraction is performed by the gradient maxima technique, while in the fourth scene, the zero crossing technique is used (see Section II - 1). All this preprocessing is independent of the stereo-matching program and could be done in a fraction of a second on specialized hardware. In its current implementation, the preprocessing requires an average of 120 seconds of C.P.U. time per image on a Sun 3

workstation for the gradient maxima technique and 20 seconds for the zero crossing technique.

VI.2 - Stereo-matching

We first present the results concerning the indoor scenes shown in Figure 8, 9, 10. For each Figure we show in (a) the original pictures and in (b) the initial edge segments. Only the segments of length greater than 12 pixels are selected by the stereo-matching program : these segments are shown in (b').

In (c) we show the pairs of segments matched by the stereo-matching algorithm. Homologous segments are labeled with the same number.

In (d) and (e) we show, superimposed on the right image segments, the horizontal distance (computed from the optical center of the camera and projected on the horizontal plane) and the elevation (computed from the floor) of the midpoints of the reconstructed 3D segments. These mesures agree very well with the direct measures we can make on the environment. Also, in (d') and (e') we show the results obtained on the same scene, but observed from a slightly different viewpoint (usually an almost horizontal translation of a few tens of centimeters and a small rotation of less than 15 degrees). It is interesting to verify the good stability of the vertical coordinate in the first and third examples, and the global small vertical translation in the office example due to a small vertical translation of the cameras. Also the measured horizontal distance allows for a qualitative verification of the validity of the 3D reconstruction.

A careful examination of the matching results showed an error rate lying between 1 % and 2 %. Most of the mistakes come from the absence of an edge in one of the images, yielding a false match which stills agree with its neighborhood (a node in the disparity graph connected to a sufficient number of other nodes). Of course, one could use the results obtained from several viewpoints to correct some of these errors. We are currently working on this improvement. Also, some of the matches correspond to specularities, which yield virtual 3D segments which can appear above the celling or bellow the ground. Additional semantic knowledge could be incorporated in the system to remove these segments. Notice however that the segments 53 and 54 in the corridor example are measured at the correct negative elevation, due to a slope one can notice in the image. Notice also the fact that in the six scenes, horizontal segments on windows,

on the ground and on celling usually form clusters of segments having the same elevation, and that the computed horizontal distance is sound with the observed scene and allows for an easy detection of the closest obstacles.

In these indoor scenes, the epipolar lines are tilted and make an angle with the horizontal plane of about 45 degrees. This is obtained by placing the second camera on the right of and below the first one. This is very interesting because it allows for the accurate reconstruction of both vertical and horizontal segments which represent the majority of the image segments.

In Figure 11, we show the results obtained on an industrial part. The geometry of the cameras was totally different (larger angle stereo, almost horizontal epipolar lines, teleobjectives instead of large angle lenses). After an automatic calibration of this new system, the same stereo-matching program was used. The results for two different positions of the object are shown in Figures 11 (a) et (d). In the vertical view of the reconstructed segments it is interesting to notice the linear clusters corresponding to points which are in a vertical plane, and to observe a horizontal rotation of about 40 degrees of this plane. This qualitative observation was confirmed by a quantitative experiment. The 3D segments were processed by the geometric matcher developed in our laboratory [AYACHE-85] and most of the midpoint segments present in the 2 scenes were matched between these two scenes by a solid displacement corresponding exactly to the correct one. The average error between 3D matched points is 4.5 millimeters, while the diameter of the object is about 25 centimeters, which gives a good idea of the quality of the reconstruction.

To conclude this experimental report we present in the following table the computing time devoted to the stereo-matching process. The program is written in a highly recursive style in C and runs on a Sun 3 workstation.

	Room	Office	Corridor	Industrial part
Number of left segments	385	362	177	334
Number of right segments	401	370	188	374
Number of generated hypotheses	75	75	75	32
Number of pairs of matched segments	202	223	86	150
CPU time, construction of the adjacency graph	1.5 s	1.5 s	0.5 s	1.0 s
CPU time, generation of hypotheses	0.2 s	0.2 s	0.1 s	0.2 s
CPU time, propagation and validation of hypotheses	5 s	5 s	2 s	3 s

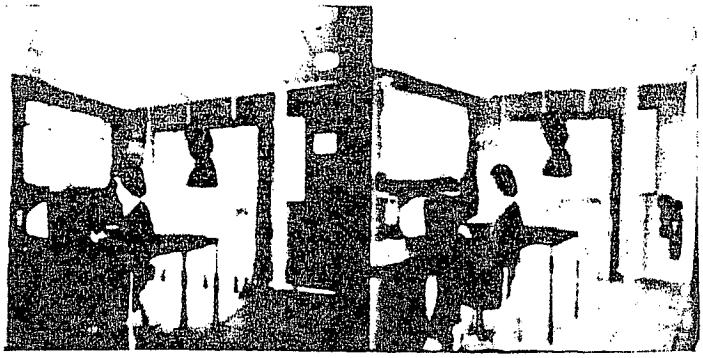


Figure 8 (a) : Stereo view of a room

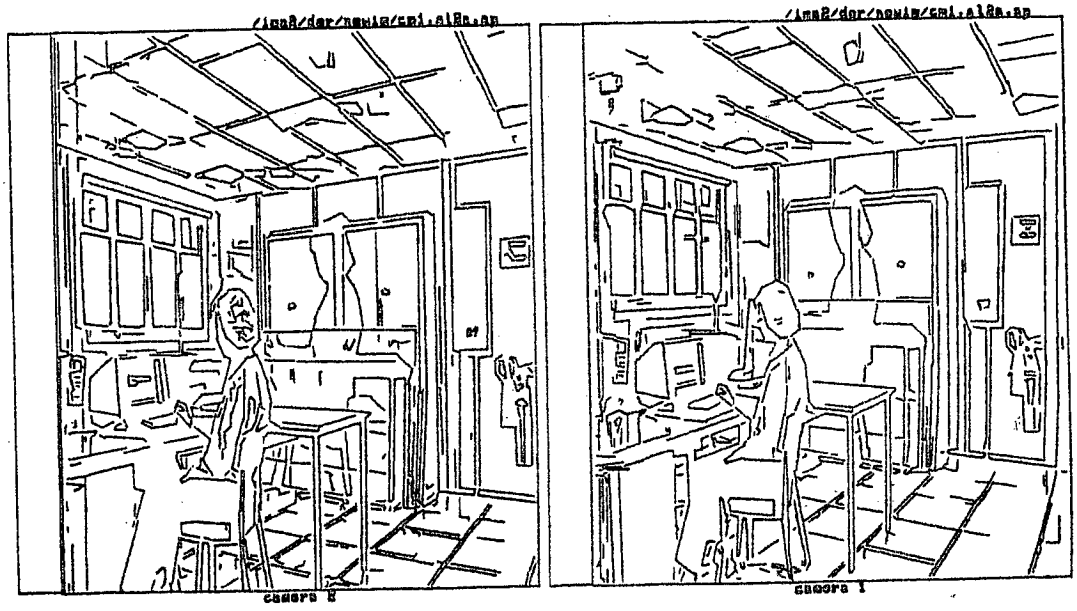
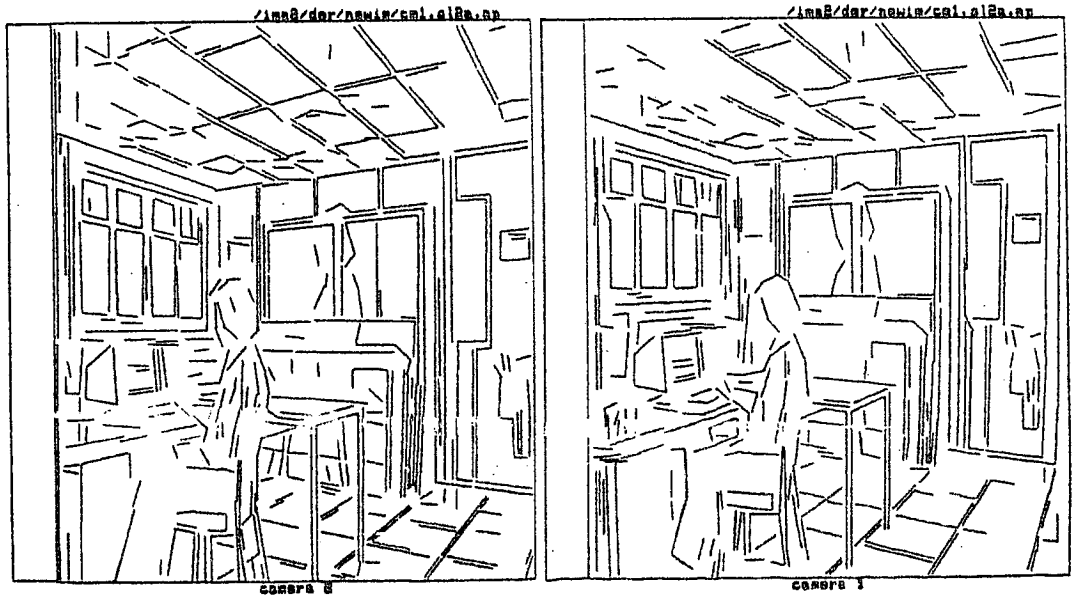


Figure 8 (b) : Initial edge segments

Figure 8 (b') : Edge segments of length greater than 12 pixels



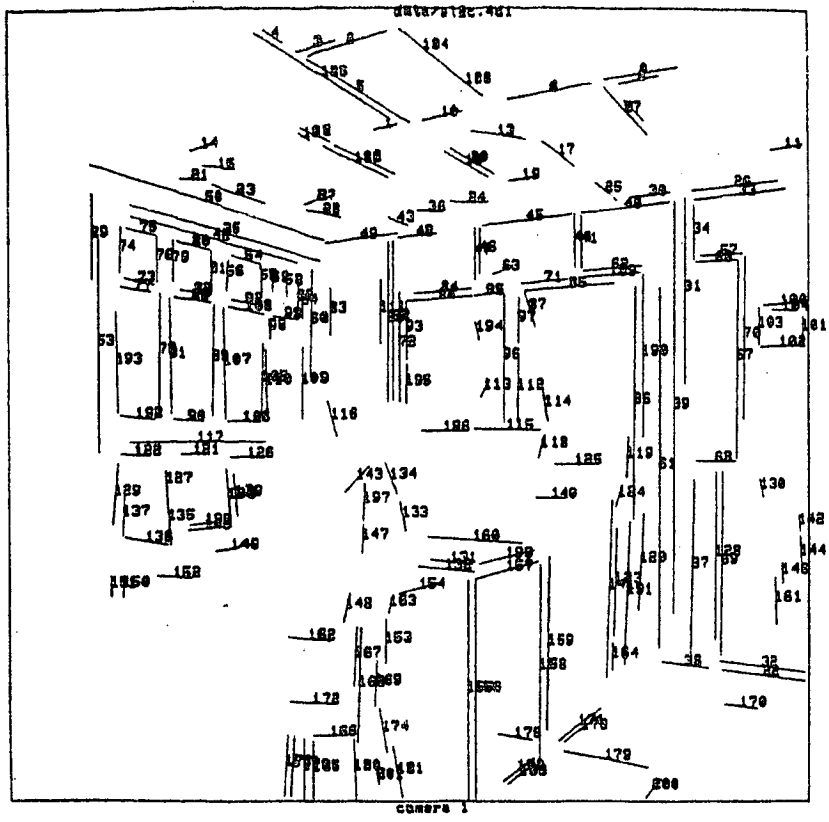
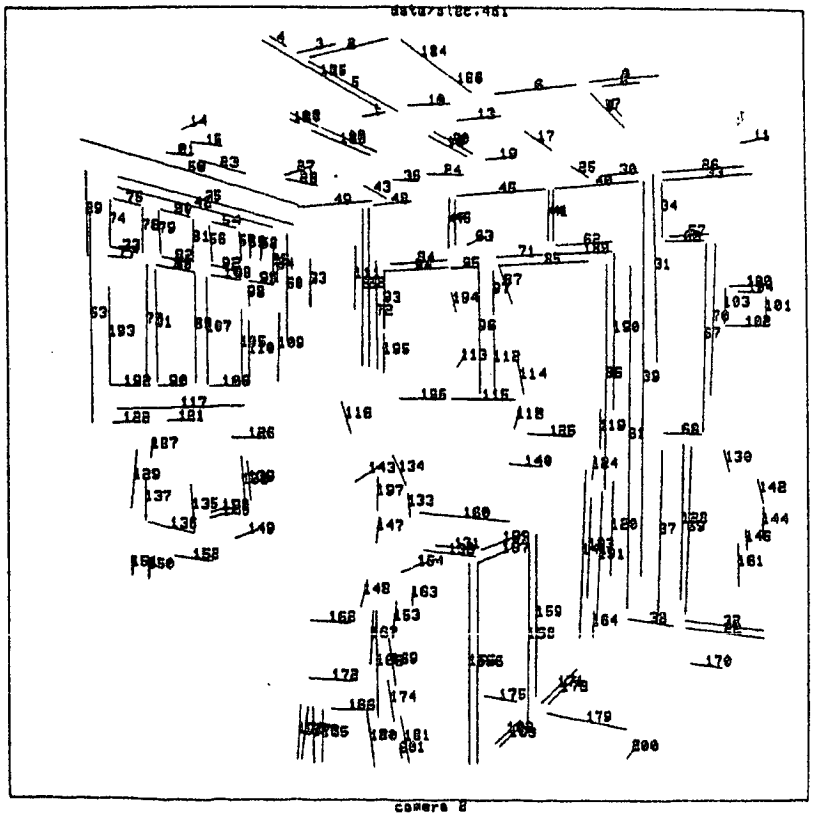


Figure 8 (c) : Matched segments (of length greater than 12 pixels)



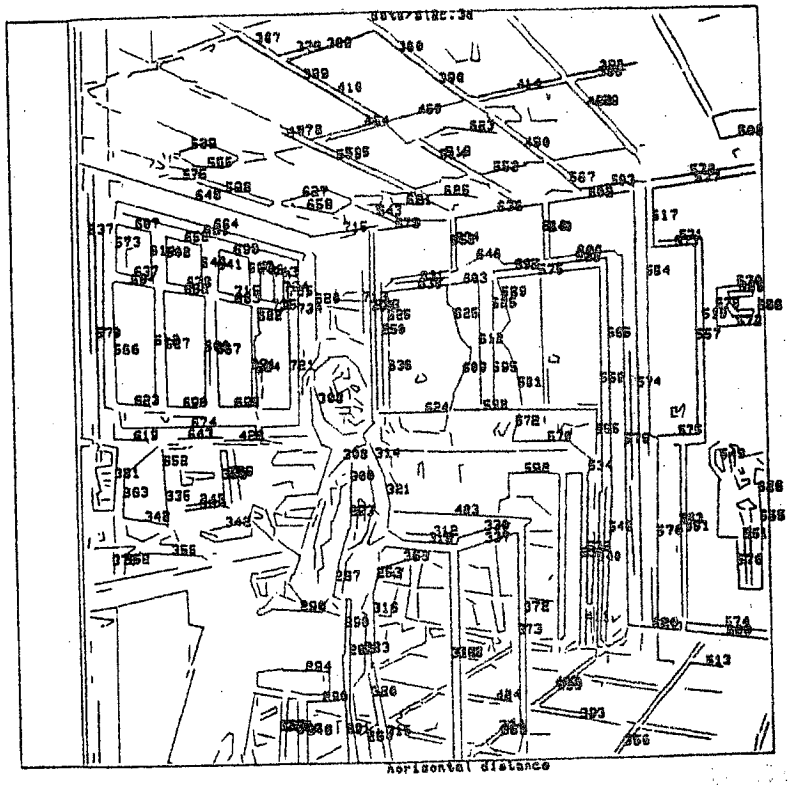
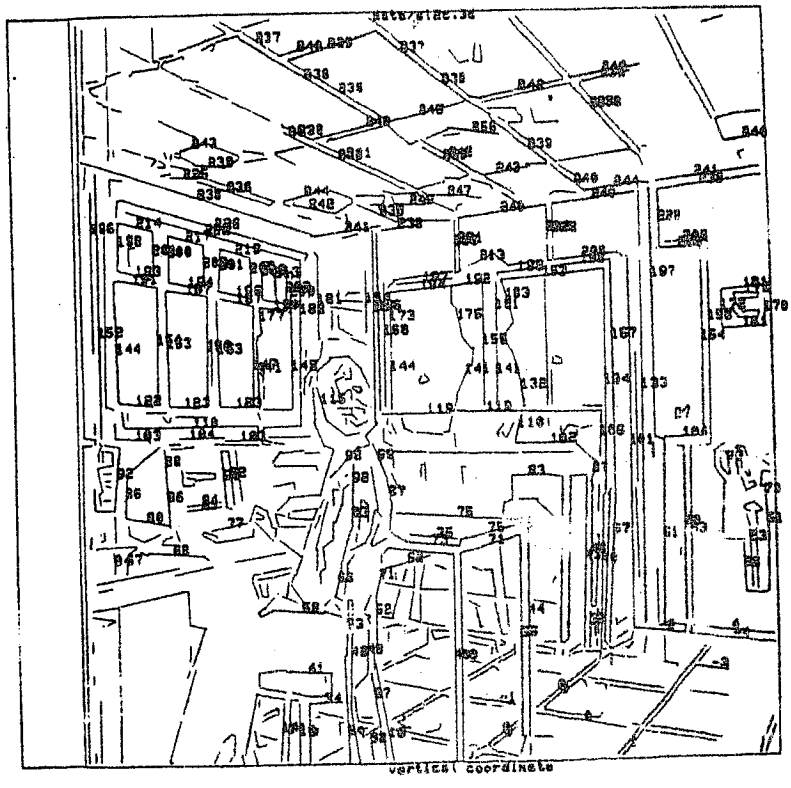


Figure 8 (d) : Horizontal distance (cm)

Figure 8 (e) : Elevation (cm)



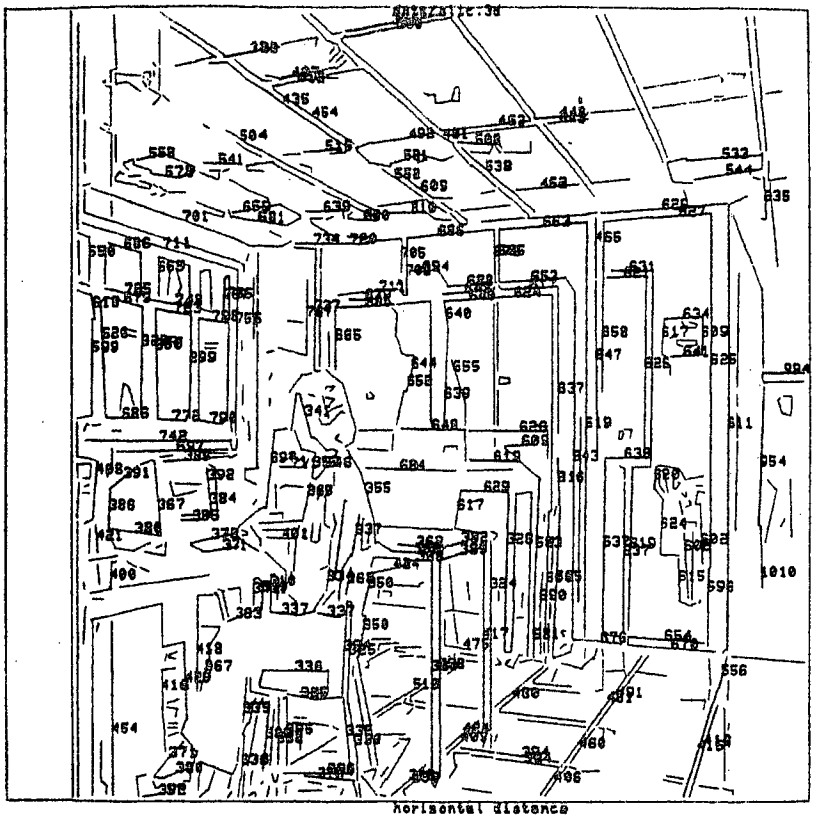
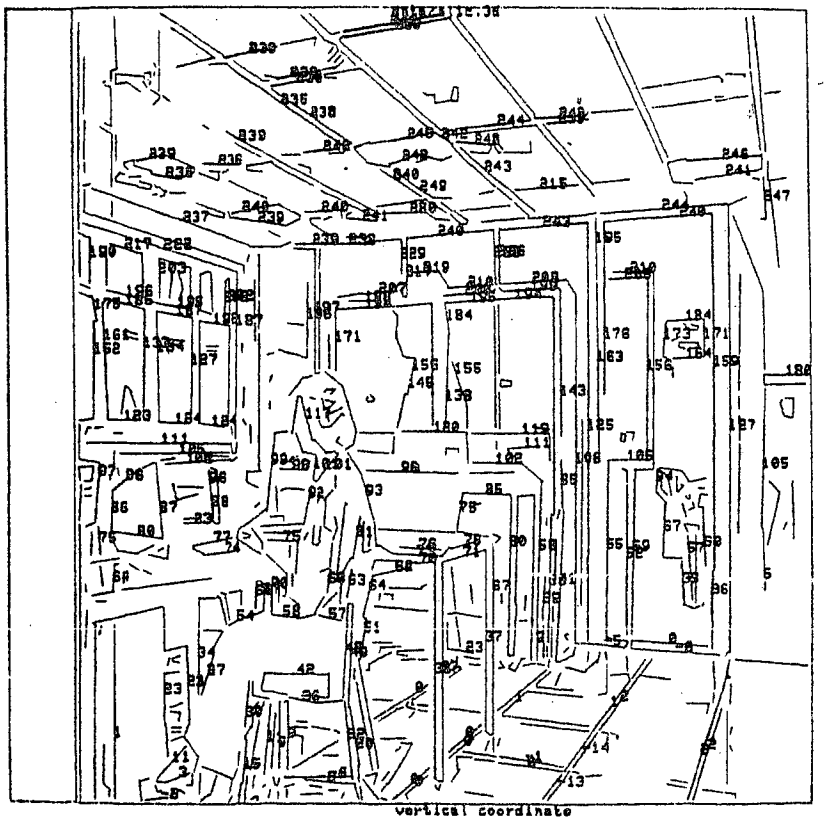


Figure 8 (d'), (e') : Horizontal distance and elevation obtained from another viewpoint



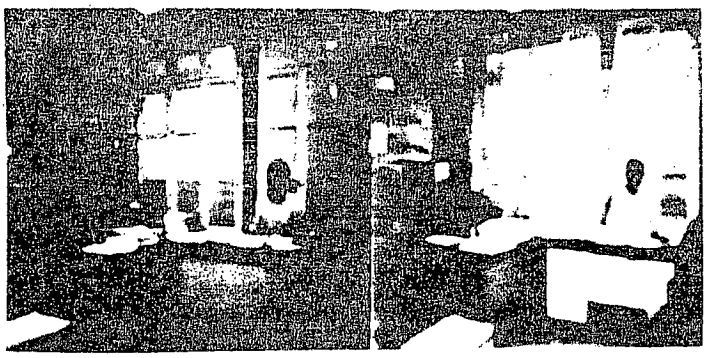


Figure 9 (a) : Stereo view of an office room

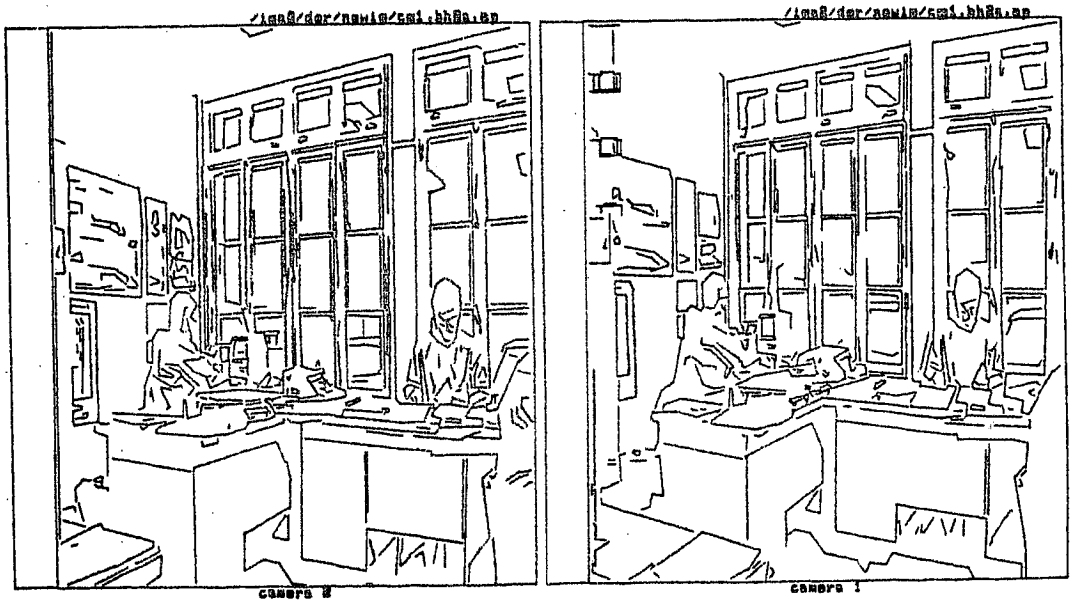
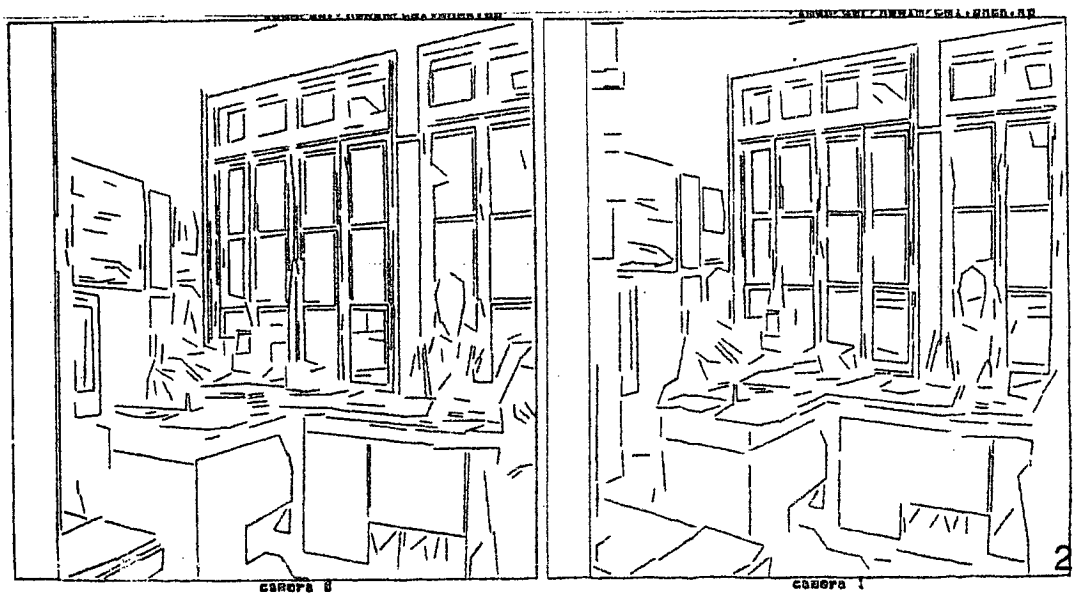


Figure 9 (b) : Initial edge segments

Figure 9 (b') : Edge segments of length greater than 12 pixels



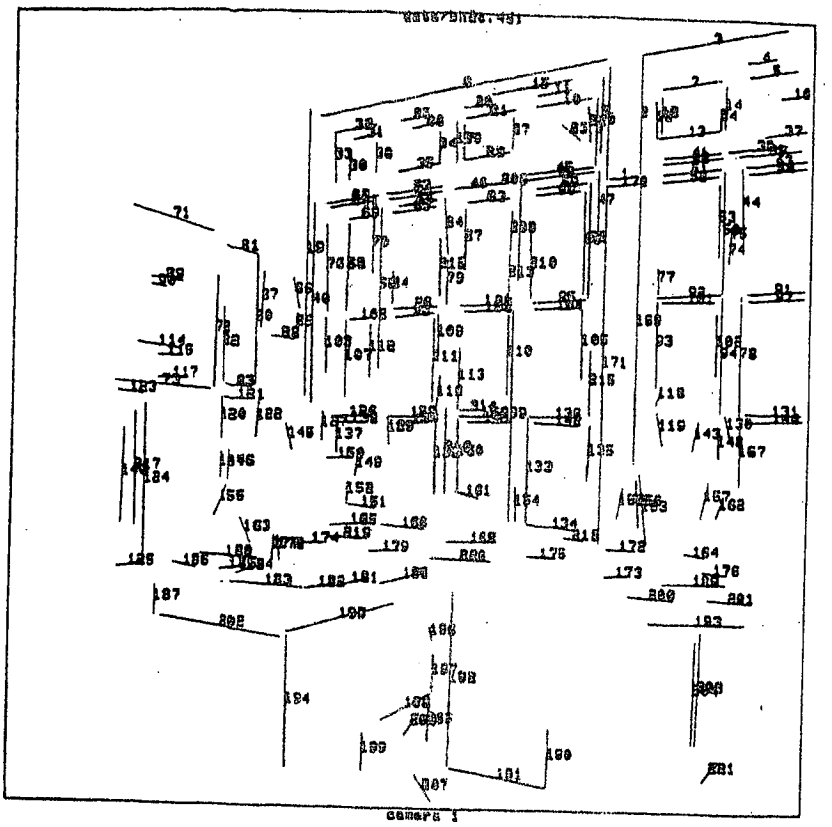
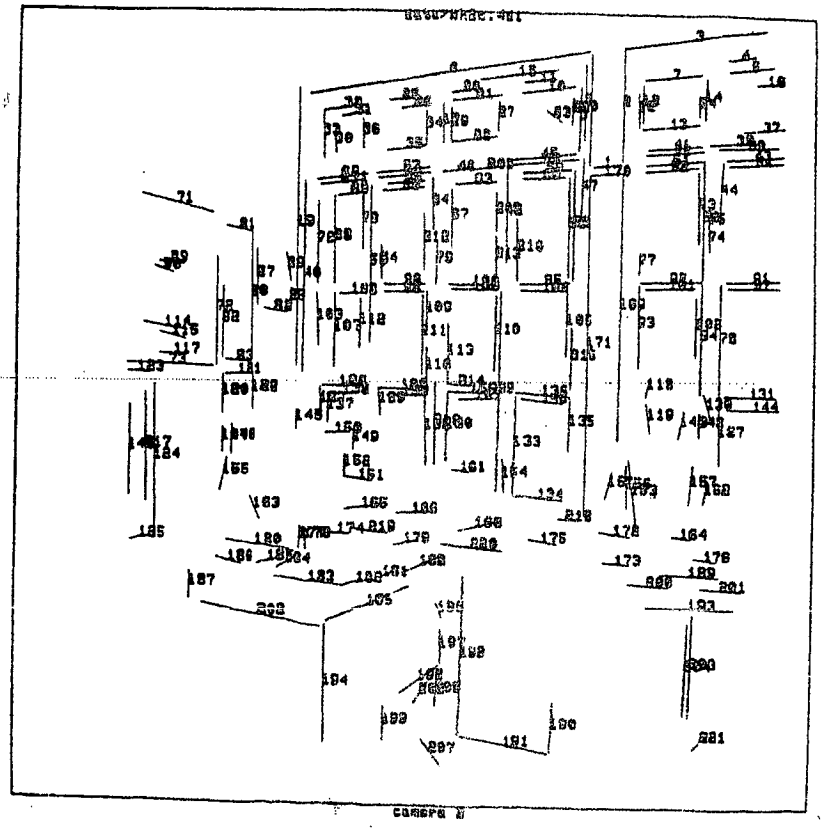


Figure 9 (c) : Matched segments (of length greater than 12 pixels)



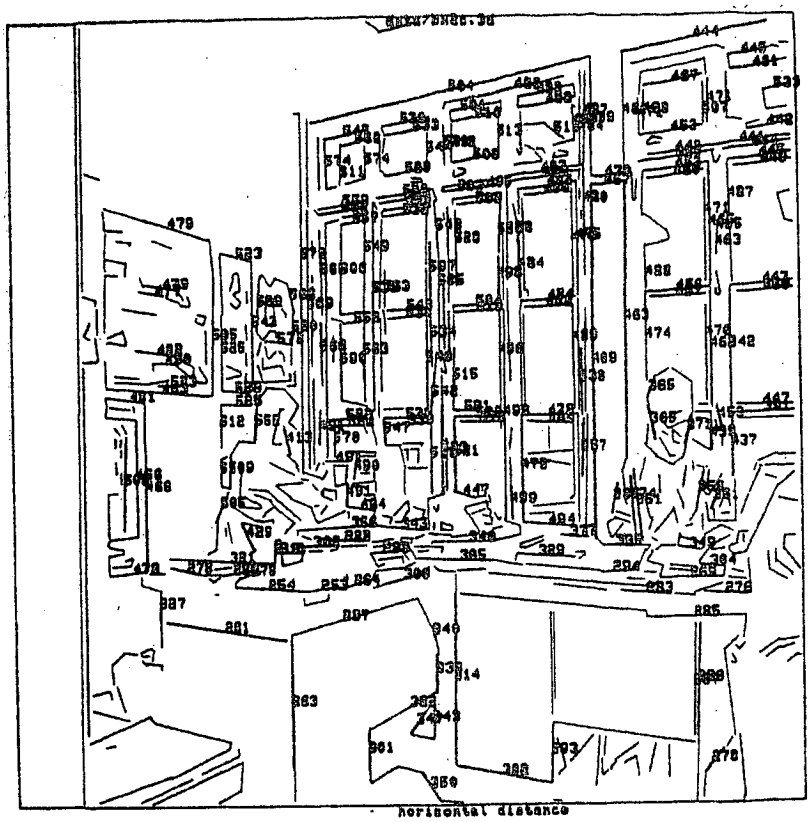
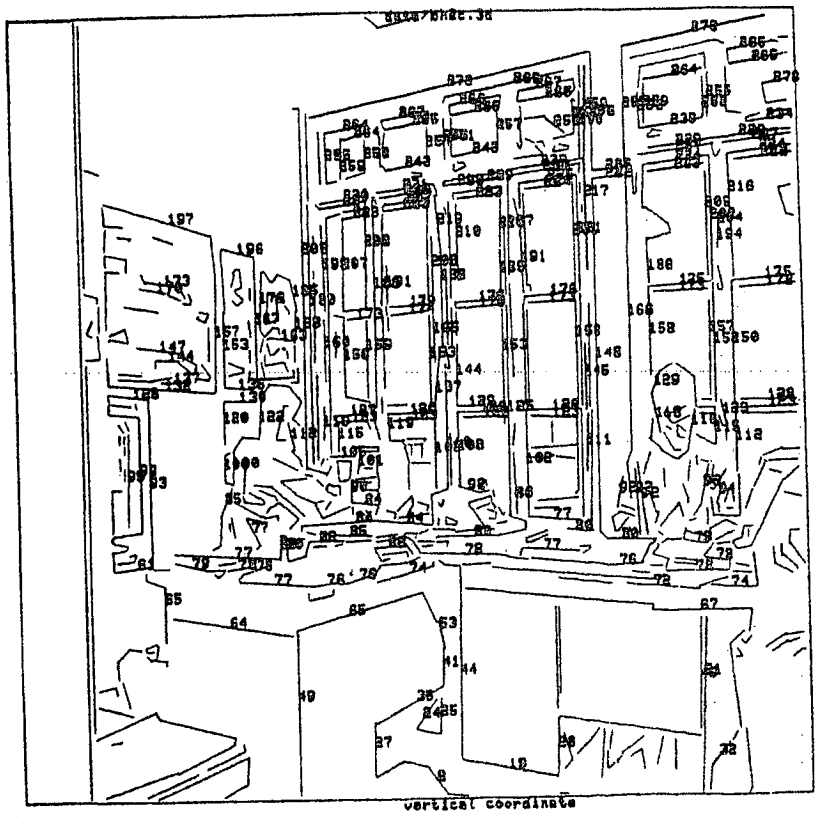


Figure 9 (d) : Horizontal distance (cm)

Figure 9 (e) : Elevation (cm)



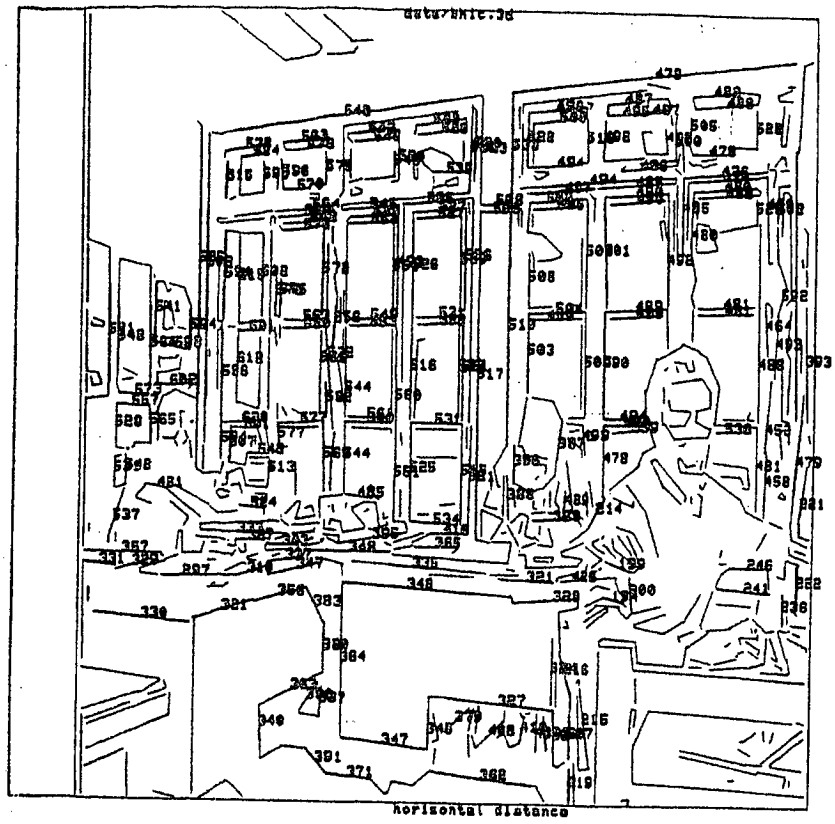
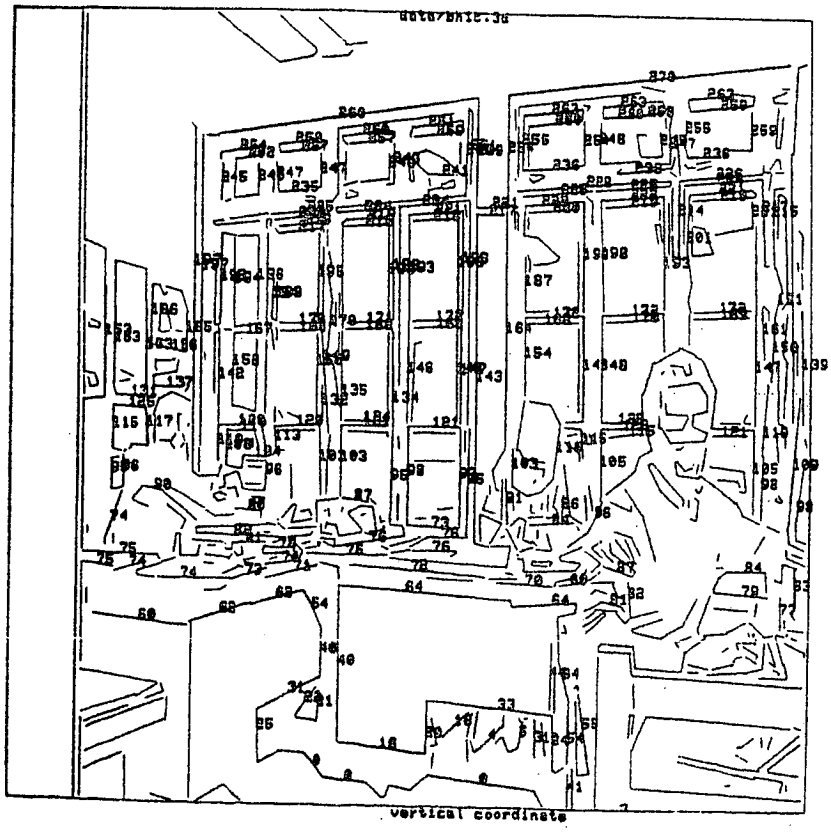


Figure 9 (d'), (e') : Horizontal distance and elevation obtained from another viewpoint



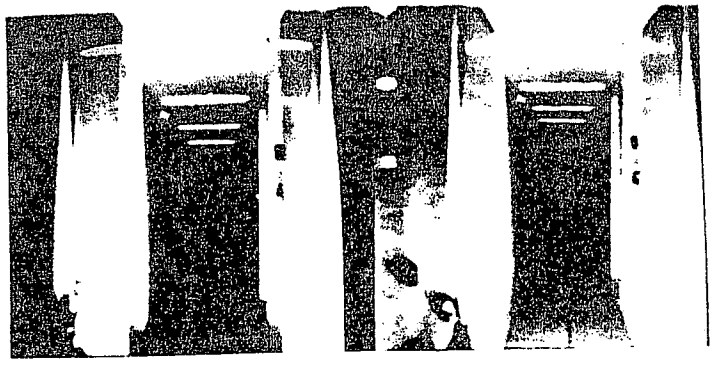


Figure 10 (a) : Stereo view of a corridor

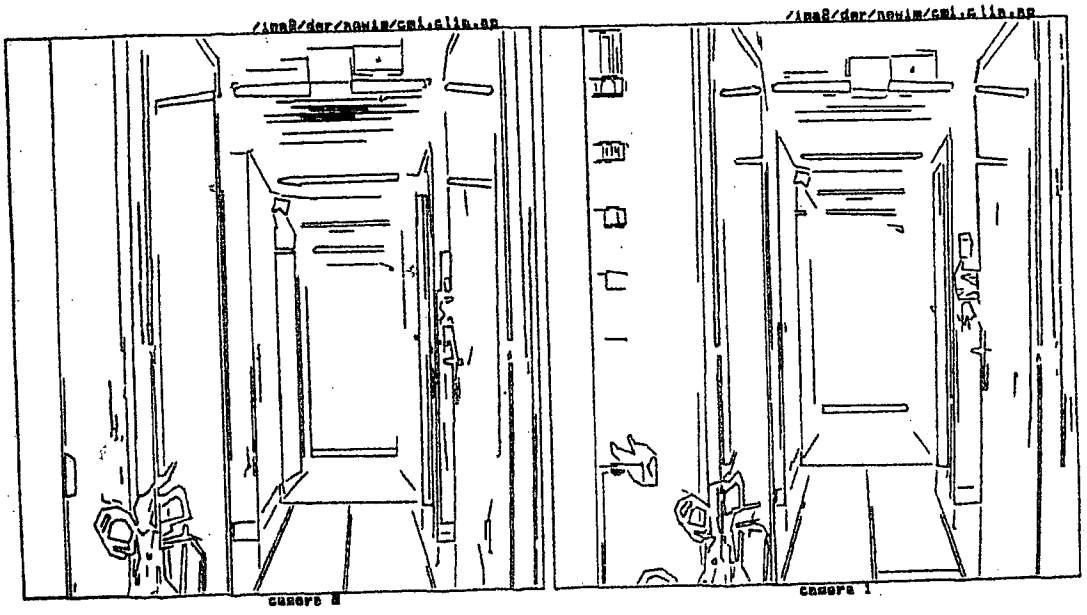
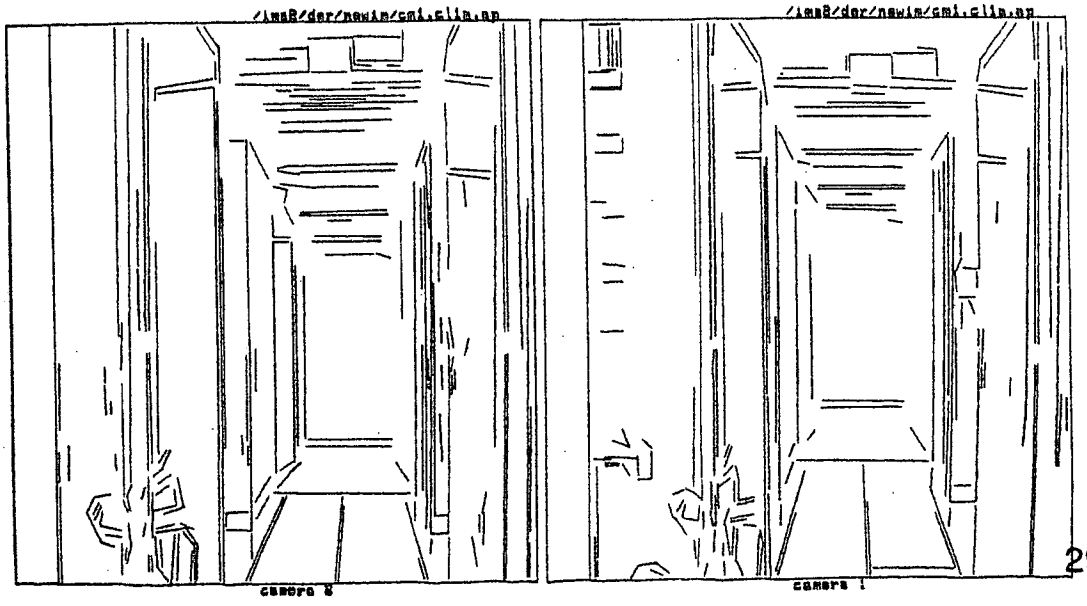


Figure 10 (b) : Initial edge segments

Figure 10 (b') : Edge segments of length greater than 12 pixels



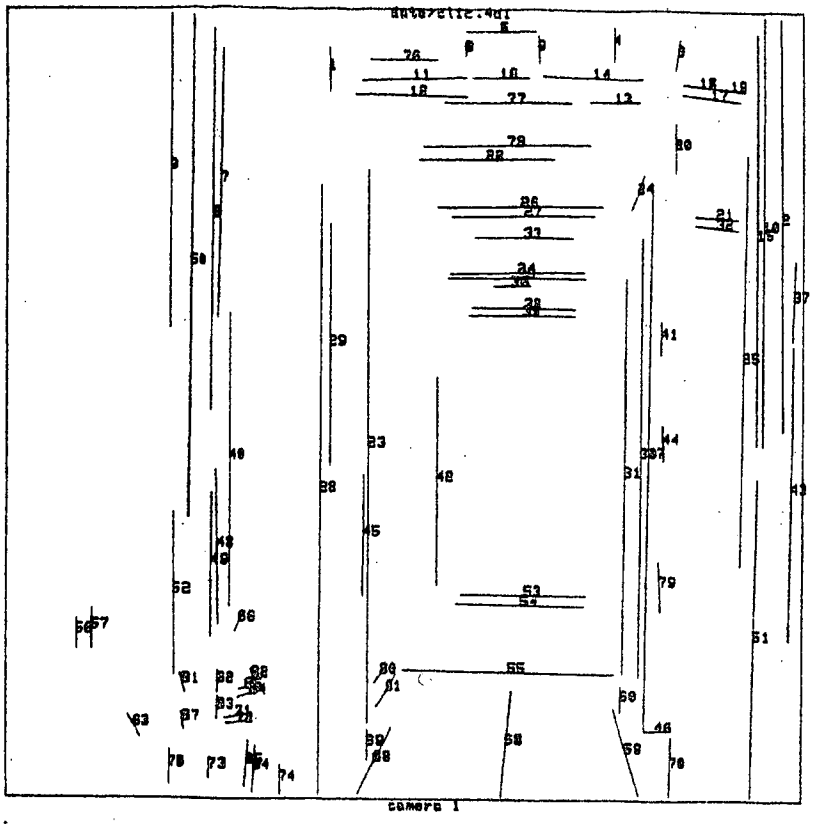
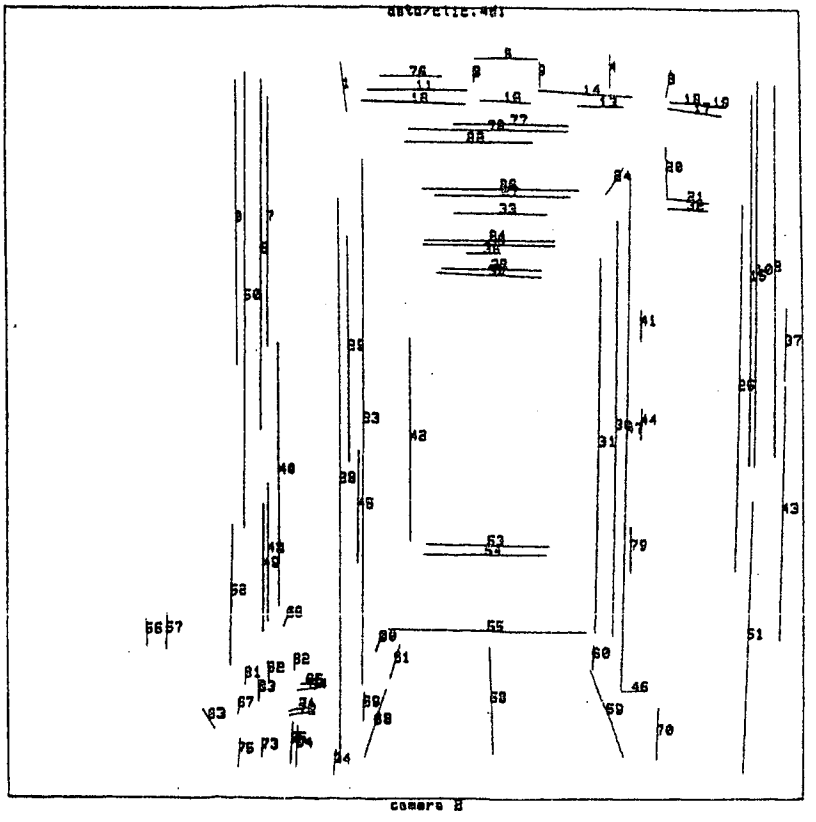


Figure 10 (c) : Matched segments (of length greater than 12 pixels)



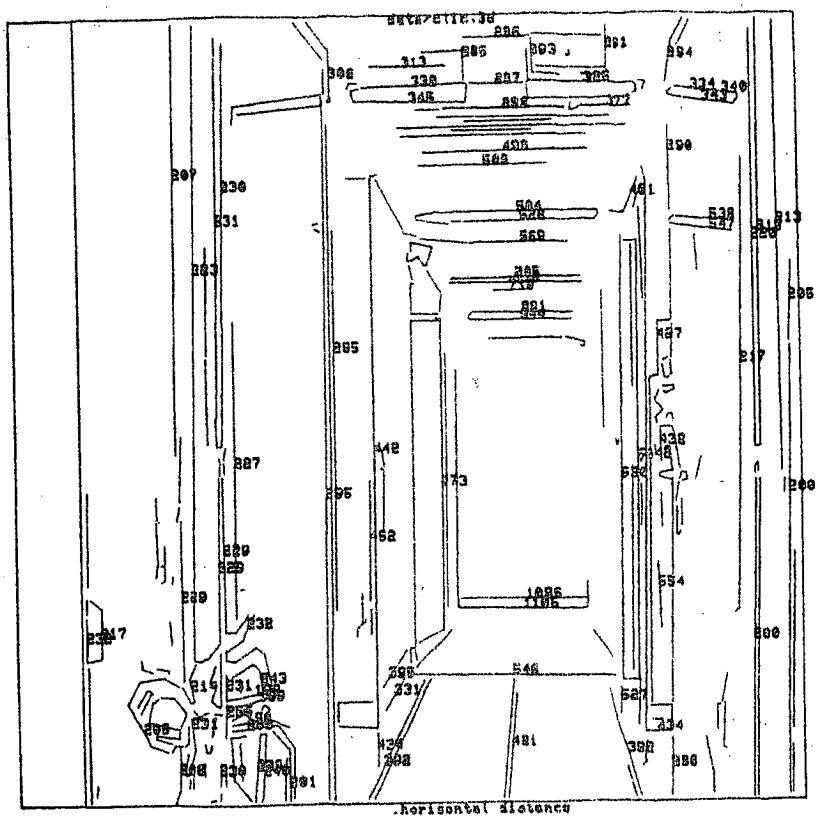
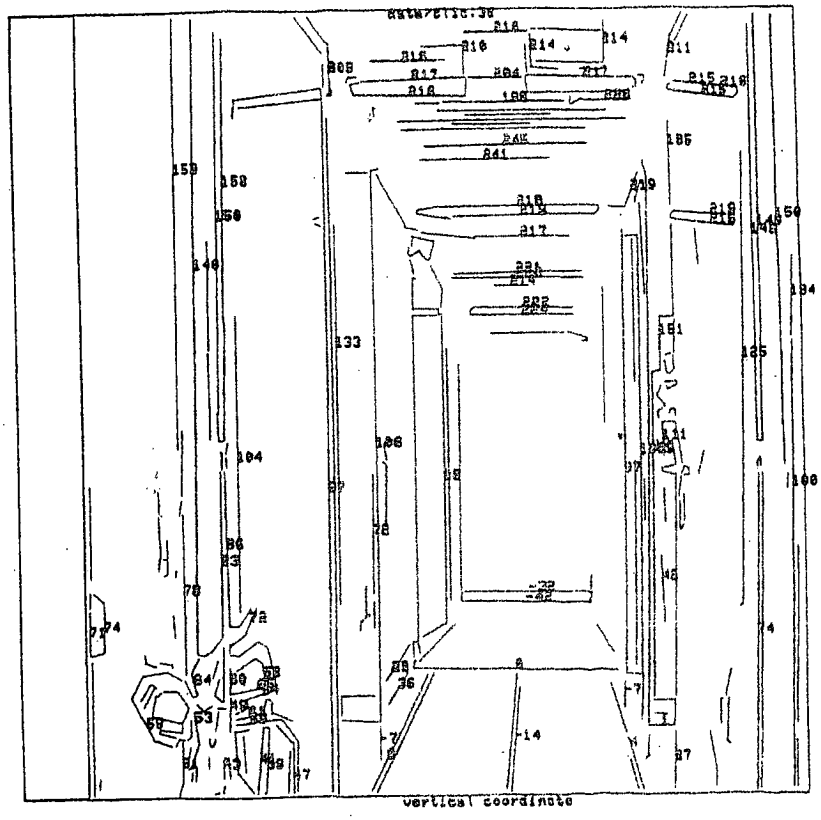


Figure 10 (d) : Horizontal distance (cm)

Figure 10 (e) : Elevation (cm)



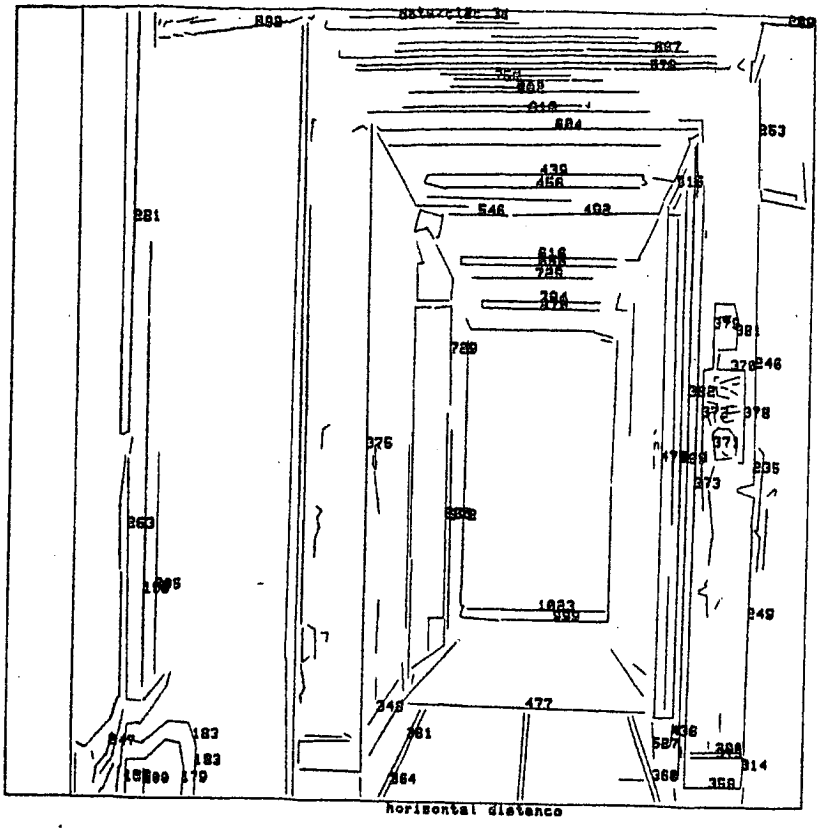


Figure 10 (d'), (e') : Horizontal distance and elevation obtained from another viewpoint

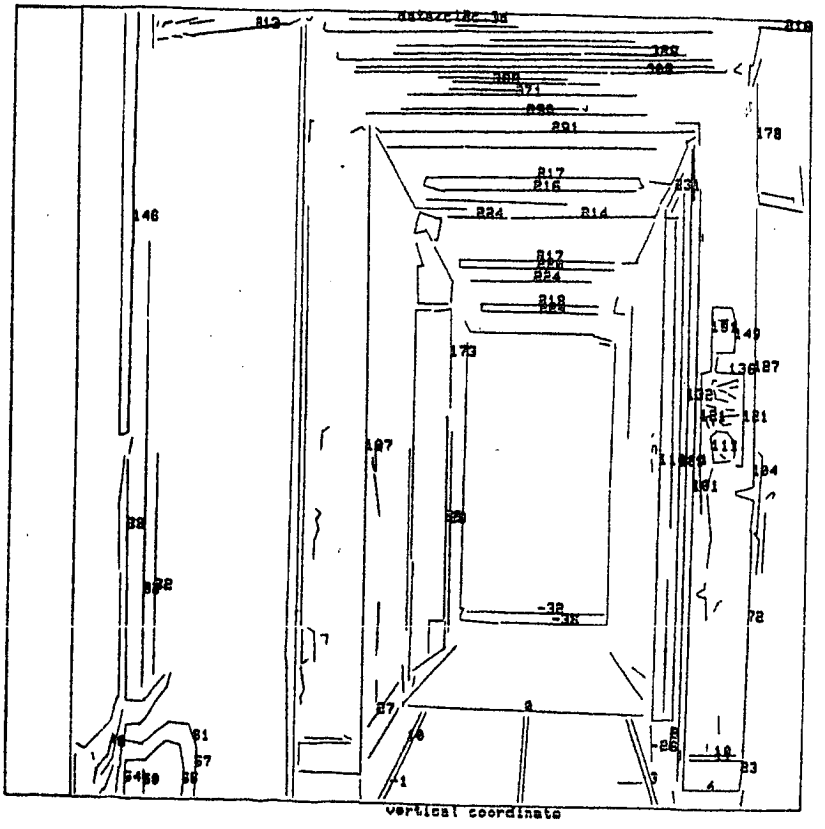


Figure 11 : 3D positioning of an industrial part

Scene S_1 Scene S_3

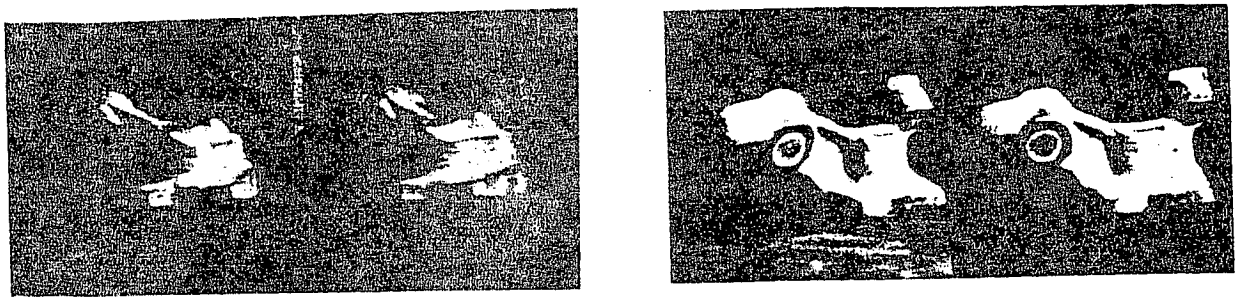


Figure 11 (a) : Initial images

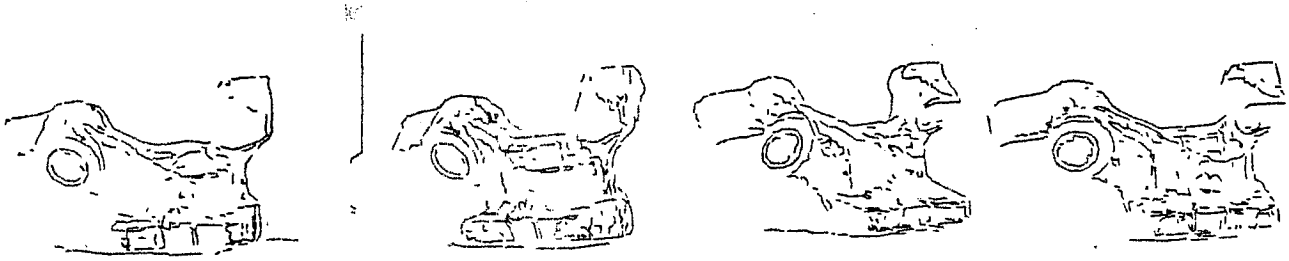


Figure 11 (b) : Stereo images and extracted contours

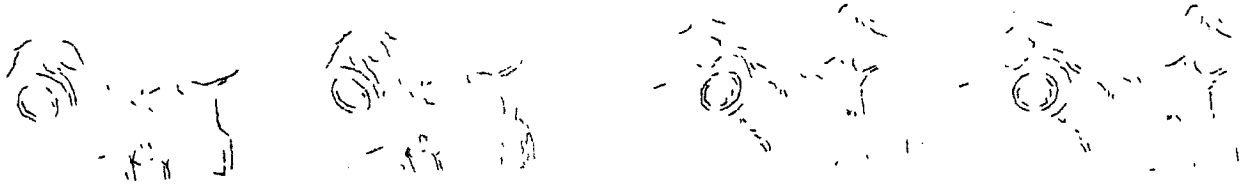


Figure 11 (c) : Segments matched by the stereo program

Figure 11 (d) : Front and vertical view of reconstructed 3D segments. The geometric matcher found the correct motion between the two scenes (see text)

VII - CONCLUSION AND DISCUSSION

We have presented a new approach to solve the stereo-matching problem using a graph based description of images and a technique of prediction and propagation of hypotheses within a disparity graph. The method appears to be particularly fast and reasonably robust.

The use of the disparity graph defining the connectivity between 2 nodes (L,R) and (L',R') each time the distance between the reconstructed 3D segments is smaller than a preset constant ε yields smooth surface patches which are *intrinsic entities* of the 3D scene, and remain unchanged if the point of view or the geometry of the steroscopic system are modified. The pre-computation of a disparity gradient as a function of the position of points in the 3D space enables efficient computations compatible with robotics applications requirements.

Robustness could be slightly improved by adding some intensity-based local features to the geometric description of segments, such as the average intensity on each side of a segment and the average contrast across a segment. The comparison of these additional features should reduce again the number of wrong hypotheses and also the number of final false matches.

The stereo matcher presented in this paper has already been used to demonstrate fast recognition and positioning of 3-D objects as well as to complete some navigation tasks for an autonomous vehicle developed in our laboratory. The results are presented in other papers [AYACHE-85], [FAUGERAS-86b].

To conclude, we think that the methodology developped in this application, that is the use of a symbolic geometric description of the images and of a graph search based on a prediction and propagation of hypotheses strategy is quite general and could probably be applied, by relaxing the epipolar constraint, to achieve fast registration between a time sequence of monocular moving images. Also, the prediction and propagation scheme is very well suited to include this kind of registration process within a closed loop, results obtained at previous stages being used to guide the prediction at the current stage. We are currently working on this problem.

VIII - ANNEX A : CALIBRATION OF THE STEREOSCOPIC SYSTEM

VIII.1 - Computation of the perspective transformations

Each camera can be modeled by a perspective transformation T. This transformation is linear in homogeneous coordinates and can be represented by a 3×4 matrix which maps scene points to image points

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = T \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad (1)$$

The perspective transformations T_1 and T_2 of the cameras are automatically estimated by the following procedure. A planar grid of precisely located orthogonal straight lines is placed in front of the two cameras in (at least 2) different known positions. Images of theses lines are extracted, and their equations are accurately computed by a least squares estimator. Then the intersection points of the grid are computed. As soon as 6 different non coplanar grid points have been registred between the scene and an image, the corresponding perspective transformation T is estimated by a least squares estimate.

In effect, each transformation T constrains 11 unknowns (and not 12 as T is defined up to a scale factor), and each registration yields two linear equations by elimination of s in system (1). In practice, more than 50 registrations are computed to obtain an accurate estimate of T.

VIII.2 - Computation of epipolar lines

Given a point I_1 in one image, its homologous point I_2 is constrained to lie within a line in the second image called the epipolar line of I_1 (see Figure A1). This line is simply the image by the second camera of the line I_1C_1, the inverse image by camera 1 of I_1. It can also be seen as the intersection of the epipolar plane $I_1C_1C_2$ with the image plane of the second camera.

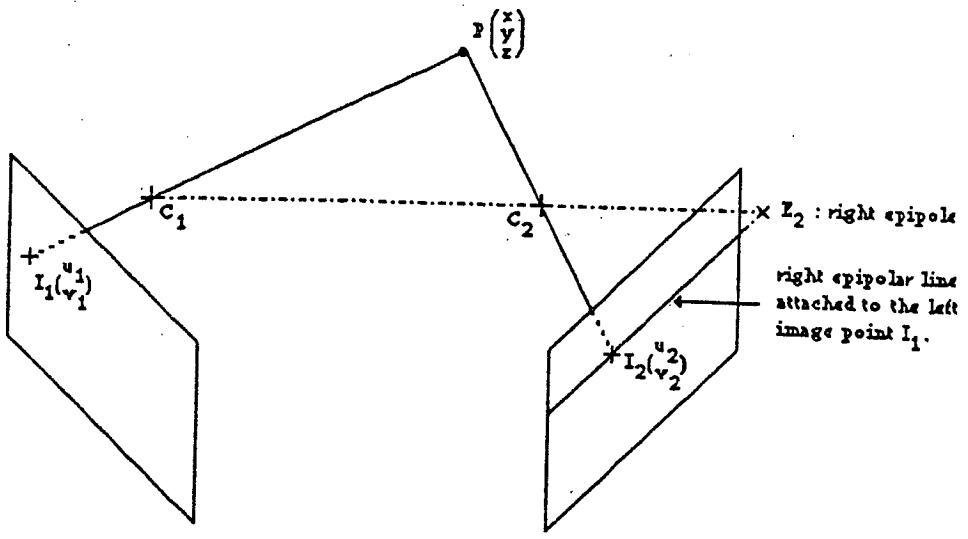


Figure A1 : Geometry of the epipolar lines in the right image

For the sake of efficiency, it is important to be able to compute easily the epipolar line attached to any image point. First, it is obvious that any epipolar line in image 2 will go through a common point E_2 called the epipole 2. E_2 is simply the image by the second camera of the optical center of the first camera. Knowing the perspective transformation T_1 of camera 1, the coordinates $[x_1, y_1, z_1, 1]^t$ of C_1 are obtained by resolving :

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = T_1 \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \quad (2)$$

Then the image coordinates $[u_{e2}, v_{e2}]$ of E_2 are obtained by applying T_2 to C_1.

$$\begin{bmatrix} su_{e2} \\ sv_{e2} \\ s \end{bmatrix} = T_2 \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \quad (3)$$

Given the position of the epipole, to determine the equation of an epipolar line, it is sufficient to compute the coordinates of a supporting vector λ_2 (see Figure A2).

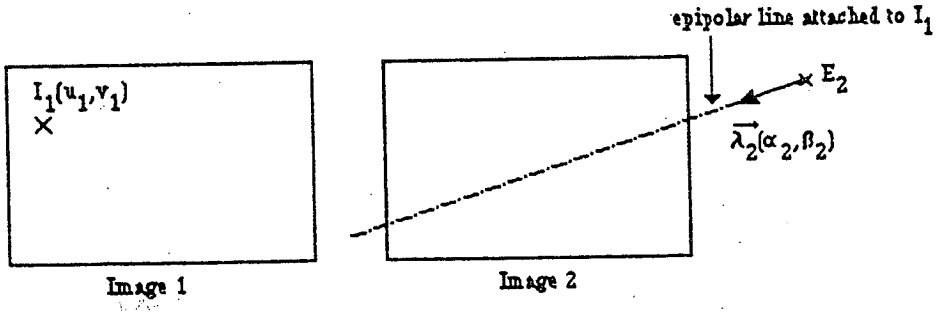


Figure A2

The remarkable point is that there is a constant linear relationship between the coordinates $(u_1, v_1, 1)$ of an image point I_1 and the coordinates (α_2, β_2) of the supporting vector λ_2 of the corresponding epipolar line :

$$\begin{bmatrix}\alpha_2\\ \beta_2\end{bmatrix} = M_{12}\begin{bmatrix}u_1\\ v_1\\ 1\end{bmatrix} \qquad (4)$$

where M_{12} is a 2×3 matrix dependent on T_1 and T_2 exclusively and independent of the position of I_1 (see [FAUGERAS-86] for details).

Of course the same formalism applies when inverting the roles of cameras 1 and 2. The 2×6 coefficients determining matrices M_{12} and M_{21} and the 2×2 coordinates of the epipoles E_1 and E_2 are computed once for all and stored at the end of the calibration procedure. During the matching procedure described in the next section, these coefficients will be used to compute in a straigthorward manner the epipolar line attached to any point either in camera 1 or in camera 2.

VIII.3 - Reconstruction of 3-D segments

When two segments S_1 and S_2 have been matched between the left and right images, it is possible to reconstruct a 3-D scene segment S of maximum length whose images S'_1 and S'_2 are strictly included within S_1 and S_2 (see Figure A3).

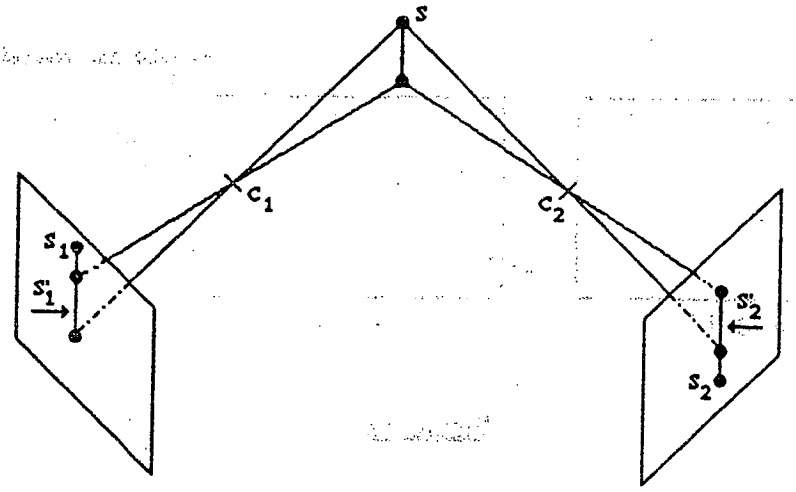


Figure A3 : Reconstruction of a scene segment from its two images

This is done by first computing the epipolar lines attached to the vertices of each segment and by computing the corresponding intersection with the other segment. If there is an intersection, the vertex and the intersection point are kept as being the homologous images of a vertex of S. When 2 such pairs of homologous image points have been found, the 3 coordinates [x, y, z] of the vertices of S are computed as follows. For a given pair (I_1, I_2) of homologous points, the following relations hold :

$$\begin{bmatrix} su_1 \\ sv_1 \\ s \end{bmatrix} = T_1 \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} s'u_2 \\ s'v_2 \\ s' \end{bmatrix} = T_2 \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (6)$$

By eliminating s and s' in (5) and (6), one ends up with 4 linear equations of the 3 unknowns [x, y, z]. In fact only 3 of these equations are linearly independent, because I_1 and I_2 have been chosen within a common epipolar plane. Therefore this system has an exact solution (x, y, z) straigthforwardly computed.

IX - ANNEX B : Computation of the disparity gradient limit

IX.1 - Definition

This definition is given for left to right matches. A symmetric definition is used for right to left matches. Given a left to right match (L,R,DISP) corresponding to a 3D point P (see Figure B1), we want to relate the variation of depth of P with the variation of disparity between L and R. To do this, depth is computed along the line of sight of the left camera. A maximum variation of depth of ε around P corresponds to the points P1 and P2 whose images in the right image are R1 and R2 respectively having disparities DISP+Δ1 and DISP+Δ2 with L. The idea is to connect a neighboring match (L',R',DISP') to (L,R,DISP) in the disparity graph if and only if the disparity gradient DISP'-DISP lies within the interval [Δ1,Δ2].

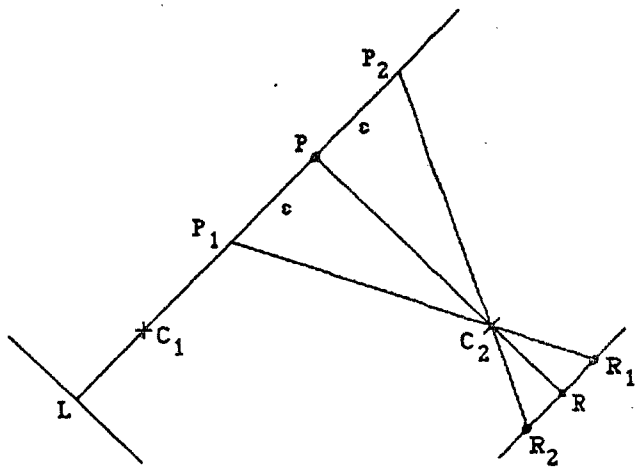


Figure B1 : Computation of the disparity gradient limit as a function of the point position

IX.2 - Practical computation

Given (L,R), we compute the coordinates of P by the technique described in section VIII-3. Then, having the coordinates of C1, we compute

$$\vec{\mu} = \vec{C_1P}/\|\vec{C_1P}\|$$
$$\vec{OP_2} = \vec{OP} + \varepsilon\vec{\mu}$$
$$\vec{OP_1} = \vec{OP} - \varepsilon\vec{\mu}$$

Finally the images of P1 and P2 are obtained by applying the perspective transformation of the right camera to P1 and P2.

To speed up the computation of $\Delta 1$ and $\Delta 2$ during the propagation process, we build a look-up table during a preliminary off-line process. We choose a point L located in the middle of each bucket in the left image: then for each such point we determine n regularly spaced potential right matches on the corresponding right epipolar line lying within the a priori potential disparity range. For each couple (L,Ri,DISPi) the values of $\Delta 1$ and $\Delta 2$ are computed and stored. During the matching process and given a current match (L,R,DISP), the program selects the bucket corresponding to L, and selects the disparities DISPi and DISPj closest to DISP. The tolerance values $\Delta 1$ and $\Delta 2$ are obtained by a simple linear interpolation. In the experiments with the indoor scenes, the values of $\Delta 1$ and $\Delta 2$ vary from a fraction of a pixel at a distance of 10 meters to 5 or 6 pixels at a distance of 1.5 meter (the value of ε was set equal to 20 centimeters).

X - AKNOWLEDGMENTS

The authors want to thank O.D. Faugeras for many helpful suggestions and comments troughout the course of this work, including among others the use of bucketing techniques. They also thank G. Toscani who developed the calibration system, G. Giraudon for his work on the extraction of edge chains, and N. Rocher for the preparation of this paper. Special thanks are due to F. Lustman for his substantial help in the developmental and experimental stages of this work.

XI - REFERENCES

[ARNOLD-80] R.D. ARNOLD, T.O. BINFORD
"Geometric Constraints in Stereo Vision",
Proc. SPIE, San Diego, 238, 1980, pp. 281-292.

[AYACHE-86] N. AYACHE and O.D. FAUGERAS
"Hyper : A New Approach for the Recognition and Positioning of 2-D objects",
IEEE Trans. on PAMI, Vol. 8, N° 1, pp. 44-54, January 1986.

[AYACHE-85] N. AYACHE, O.D. FAUGERAS, B. FAVERJON & G. TOSCANI
"Matching Depth Maps Obtained by Passive Stereovision",
Third Workshop on Computer Vision: Representation and Control, Bellaire, Michigan, October 1985.

[BAKER-81] H. BAKER, T.O. BINFORD
"Depth from Edge and Intensity Based Stereo",
7th Int. Joint Conf. on A.I., pp. 631-636, Vancouver, Canada, August 1981.

[BARNARD-80] S.T. BARNARD, W.B. THOMPSON
"Disparity Analysis of Images",
IEEE Trans. on PAMI, Vol. 2, N[4, pp. 333-340, July 1980.

[BENARD-83] M. BENARD
"Extracting 3-D Coordinates of an Object from a Digital Stereopair : an Automated Method",
EURASIP 1983, Erlangen, W. Germany, pp. 227-230.

[BEN RHOUMA-83] K. BEN RHOUMA et al.
"A K2-D Perception Approach for Assembly Robots",
EURASIP 1983, Erlangen, W. Germany, pp. 227-230.

[BERTHOD-84] M. BERTHOD, P. LONG
"Stereo Viewing by Graph Matching",
(in French), pp. 1011-1016, Premier Colloque Image, Biarritz, France, May 1984.

[BERTHOD-85] M. BERTHOD
"Polygonal Approximation of edge chains",
INRIA Internal Report, 1985.

[CANNY-83] J.F. CANNY
"Finding Edges and Lines in Images",
MIT AI Lab., TR-720, 1983.

[CASTAN-84] S. CASTAN, J. SHEN
"A Stereo Vision Algorithm taking into account the Perspective Distorsions",
7th ICPR, Montreal, Canada, July 1984, pp. 444-446.

[DERICHE-86] R. DERICHE
"Optimal Edge Detection Using Recursive Filtering",
INRIA Internal Report, Jan. 1986.

[FAUGERAS-86] O.D. FAUGERAS, G. TOSCANI
"The Calibration Problem for Stereo",
CVPR'86, June 1986, Miami.

[FAUGERAS-86b] O.D. FAUGERAS, N. AYACHE, B. FAVERJON, F. LUSTMAN
"Building Visual Maps by Combining Noisy Stereo Measurements",
IEEE Int. Conf. on Robotics and Automation, April 7-10, 1986, San Fransisco, Vol. 3, pp. 1433-1438.

[GENNERY-79] D.B. GENNERY
"A Stereovision System for an Autonomous Vehicle",
Proc. 5th IJCAI, Cambridge, MA, Aug. 1977, pp. 576-582.

[GRIMSON-81] W. Eric L. GRIMSON
"From Images to Surfaces",
MIT Press, 1981.

[GRIMSON-84] W. Eric L. GRIMSON
"Computational Experiments with a Feature Based Stereo Algorithm",
A.I. Memo 762, MIT, 38 pages, January 1984.

[HWANG-82] J.J. HWANG, E.L. HALL
"Matching of Featured Objects Using Relational Tables from Stereo Images",
CGIP, Vol. 20, pp. 22-42, 1982.

[KESKES-82] N. KESKES, A. BOULANOUAR, O.D. FAUGERAS
"Application of Images Analysis Techniques to Seismic Data",
ICASSP, May 1982, Paris, pp. 855-857.

[KNUTH-75]
"The Art of Computer Programming",
Vol. 3, "Sorting and Searching",
Addison Welsey, Reading, Ma., 1975.

[LEVINE-73] M.D. LEVINE et al.
"Computer Determination of Depth Maps",
CGIP, 2, 1973, pp. 131-150.

[MARR-76] D. MARR, T. POGGIO
"Cooperative Computation of Stereo Disparity",
Science 194, pp. 283-287, 1976.

[MARR-80] D. MARR, E. HILDRETH
"Theory of Edge Detection",
Proc. Royal Society, London, B 207,1980.

[MAYHEW-81] J.E.W. MAYHEW & J.P. FRISBY
"Computational and Psychophysical Studies towards a Theory of Human Stereopsis",
Art. Intell. 17, p349-386, 1981.

[MEDIONI-83] G. MEDIONI, R. NEVATIA
"Segment Based Stereo Matching",
Proc. of the Image Understanding Workshop, June 1983, Arlington, Va., U.S.A., pp. 128-136.

[MORAVEC-79] H.P. MORAVEC
"Towards Automatic Visual Obstacle Avoidance",
Proc. 5th IJCAI, Cambridge, MA, U.S.A., Aug. 1977, pp. 584.

[MOHR-84] R. MOHR, W. WROBEL
"La correspondance en stéréovision vue comme une recherche de chemin optimal",
(in French), 4ème Congrès AFCET-INRIA de RFIA, Janvier 1984, Vol. 2, pp. 71-79.

[MORI-73] K. MORI, M. KIDODE, H. ASADA
"An Iterative Prediction and Correction Method for Automatic Stereocomparison",
CGIP, 2, 1973, pp. 393-401

[NEVATIA-76] R. NEVATIA
"Depth Measurement by Motion Stereo",
CGIP, 5, 1976, pp. 203-214.

[NISHIHARA-84] H.K. NISHIHARA
"PRISM, A Practical Real-Time Imaging Stereo Matcher",
A.I. Memo 780, MIT, 31 pages, May 1984.

[OHTA-85] Y. OHTA, T. KANADE
"Stereo by Intra- and Inter-Scanline Search",
IEEE Trans. on PAMI Vol. 7, N° 2, pp. 139-154, 1985.

[PAVLIDIS-77] T. PAVLIDIS
"Structural Pattern Recognition",
Springer-Verlag, New York, 1977.

[POGGIO-84] G. POGGIO, T. POGGIO
"The analysis of stereopsis",
Ann. Rev. Neurosci. 1984, 7 : 379-412

[POLLARD-86] S. POLLARD,J. PORRIL, J. MAYHEW, J. FRISBY
"Disparity Gradient, Lipschitz Continuity, and Computing Binocular Correspondence",
University of Sheffield, Internal Report N° AIVRU-010, 1986.

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique